Silesian University of Technology

Faculty of Automatic Control, Electronics and Computer Science

Developing a system of automatic identification of cellular subpopulations in data from single-cell mass cytometry with the use of algorithms for grouping of high dimensional data

Doctoral Dissertation

by

Aleksandra Suwalska

Supervisor

Professor Joanna Polańska, PhD, DSc

2023

Gliwice, Poland

I would like to express my most profound appreciation to my supervisor, Professor Joanna Polanska, for giving me this opportunity and her patient guidance and advice for many years.

I also wish to thank my friends from the Department of Data Science and Engineering, for the great atmosphere, for exchanging ideas, and for any help.

Special thanks to my colleague, Wojciech Prazuch, for all the conversations in the most challenging moments at the University.

The work would not have been possible without the constant support of the closest to me, my family, my best friends.

Content

Abstract

The development of single-cell technologies like mass cytometry allowed for an in-depth understanding of cellular processes and the discovery of new cell subpopulations and their roles in the organism. The knowledge can help invent new therapies, drugs, and disease prevention methods. Mass cytometry enables the simultaneous measurement of dozens of markers used for cell identification, resulting in a matrix of expression values for each cell and feature. However, the high dimensionality of single-cell datasets makes the analysis a challenge.

Tuberculosis, an infectious disease caused by the bacteria *Mycobacterium Tuberculosis*, kills millions of people every year around the globe. The medications are expensive, especially when the germs resist one or more primary drugs. The outbreak of the COVID-19 pandemic has led to the loss of recent years' progress in combating the spread of Tuberculosis. Scientists are trying to reduce the occurrence of newly diagnosed cases to a minimum by working on new therapies and drugs using, among others, mass cytometry technology.

The dissertation thesis focuses on the analysis of high-dimensional mass cytometry datasets. Based on the publically available ones, the proposed analysis pipeline tries to solve problems occurring with the existing methods. The implemented algorithms are used to process the Tuberculosis dataset provided by partners from Stellenbosch University, RPA. The analysis includes a new technique that is fully automated and reproducible for pre-gating mass cytometry events. In addition, different methods for batch effect correction are compared in terms of removing the technical variance and influencing the cell-type identification results. The proposed machine learning technique of cell-type identification considers the existence of heterogeneity of cell groups during model evaluation which is a novel approach. Furthermore, introducing the expanded feature space and two-step clustering technique allows for obtaining well-defined and separated clusters.

The analyzes carried out indicate the potential of the introduced methods in the identification of cell types and the appropriate verification of their results.

Streszczenie

Rozwój technologii badających pojedyncze komórki, jak cytometria masowa, pozwolił na dogłębne zrozumienie procesów komórkowych i odkrycie nowych subpopulacji oraz ich roli w organizmie. Wiedza ta ma szansę przyczynić się do rozwoju nowych terapii, leków i metod zapobiegania różnym chorobom. Cytometria masowa umożliwia jednoczesny pomiar dziesiątek markerów wykorzystywanych do identyfikacji typów komórek czego wynikiem jest macierz zawierająca wartości ekspresji dla każdej komórki i markera. Jednakże wysoka wymiarowość danych sprawia, że ich analiza jest wyzwaniem.

Gruźlica, czyli infekcja spowodowana przez bakterie *Mycobacterium Tuberculosis*, każdego roku zabija miliony ludzi na świecie. Leczenie jest drogie, zwłaszcza gdy bakterie uodpornią się na jeden lub więcej głównych leków. Wybuch pandemii COVID-19 doprowadził do regresji w postępach ostatnich lat w powstrzymywaniu roznoszenia się gruźlicy. Naukowcy próbują zmniejszyć występowanie nowo zdiagnozowanych przypadków do minimalnego poziomu poprzez pracę nad nowymi terapiami i lekami, wykorzystując, między innymi, cytometrię masową.

Rozprawa doktorska skupia się na analizie wysokowymiarowych danych z cytometrii masowej. Publicznie dostępne zbiory danych posłużyły do zaproponowania schematu analizy, który rozwiązuje problemy występujące przy istniejących rozwiązaniach. Zaimplementowane metody zostały wykorzystane do przetworzenia zbioru danych z badań nad gruźlicą, zapewnionego przez partnerów naukowych z Stellenbosch University, RPA. Analiza obejmuje nową, w pełni automatyczną metodę do wstępnego bramkowania danych z cytometrii masowej. Dodatkowo, istniejące rozwiązania do korekty efektu paczki zostały porównane pod względem usuwania technicznej wariancji i wpływu na końcowe wyniki identyfikacji subpopulacji. Zaproponowana technika do identyfikacji typów komórek bierze pod uwagę istnienie heterogeniczności grup komórek podczas ewaluacji modelu, co jest podejściem zupełnie nowym. Wprowadzenie rozszerzonej przestrzeni cech i dwukrokowej techniki klasteryzacji pozwala uzyskać dobrze zdefiniowane i spójne populacje.

Przeprowadzone analizy wskazują na potencjał wprowadzonych metod w identyfikacji typów komórek i odpowiedniej weryfikacji ich wyników.

1 Introduction to the mass cytometry data analysis

1.1 Motivation

With the increasing popularity and development of new single-cell technologies and computational methods over recent years, it becomes possible to investigate the heterogeneity of tissues and cell populations that may help understand better the mechanisms leading to various diseases. The knowledge about the cell state, its dynamics, and regulatory mechanisms is valuable during the invention of new treatments and the improvement of the existing ones. Despite the wealth of information provided by high-throughput technologies, analyzing the data is challenging [1]. The main characteristic of single-cell datasets is big dimensionality that usually needs high-performance computing clusters and adjusted algorithms. They are often compared to the reference proposed by experts to evaluate the analysis results, but the annotations are manual and, therefore, biased. Moreover, because many biological functions and compositions of cells are not yet understood, there is a lack of consensus about rare cell subpopulations and their natural meaning, especially in further subdivisions [1]–[3].

Tuberculosis, as one of the leading causes of death worldwide, is still a challenge and a health security threat. Studies involving Tuberculosis patients' cells may help develop new treatments and prevention methods. The scientific partners to the doctoral thesis provided a high-dimensional mass cytometry dataset involving three patient groups and over ten mln cells. Such an extensive dataset makes the analysis difficult, even using the best and state-of-the-art methods for similar problems.

Before identifying cell populations, which is the main interest in single-cell mass cytometry data analysis, the data must be appropriately preprocessed. Data acquisition from laboratory samples carries a lot of artifacts and unwanted records. The process of data cleaning is usually performed manually, therefore, is prone to human error. Additionally, preparing and processing samples without introducing technical variance to the data requires well-planned experiments and much effort; still, success is not guaranteed. The limited correction methods for the artificially introduced variance in mass cytometry datasets were imperfect for a long time. Recently some new techniques were proposed with different requirements and conditions related to their use. However, it is unclear which of these methods will be the best and whether the kind of correction impacts the further analysis results.

Finally, the cell population identification task lacks the ability to discover rare cell types (composed of less than 1% of the dataset cells). The existing solutions often depend on the manual experts' annotations to which the algorithms are compared. The knowledge about the heterogeneity of cell types is rarely considered when evaluating the model's performance. Together with the uncovered understanding of the biological functions and importance of rare cell populations, the results are considered to be worse if they do not match the known cell labels. It leads to underrated solutions that, in fact, capture the natural heterogeneity of samples.

From a technical point of view, cell-type identification is an unsupervised task that engages clustering techniques in searching for patterns and relationships within the data. The main challenge in clustering methods is determining the optimal number of clusters and scalability to big datasets. The best solution would be data-driven without the need for tuning too many parameters and specifying the number of groups. Also, the algorithm should work for high-dimensional datasets in a reasonable time.

Since our team has a successful experience with similar studies, we think we can significantly contribute to solving the abovementioned problems.

1.2 Aim of the work

The main aim of the work was to propose a solution for the problem of clustering high-dimensional datasets containing small populations of observations. The proposed algorithm should be able to process datasets containing millions of cells and dozens of markers with high sensitivity to rare cell types (small subpopulations). The method should also consider the heterogeneity of biological data and potential problems resulting from the sample preparation in a laboratory. Therefore, the subordinate goal of the doctoral thesis was to examine the data preprocessing steps dedicated to mass cytometry data and propose improvements in the process.

The main focus of the data processing section was the pre-gating step and the correction of the technical variance introduced to the samples. The pre-gating is a primary data-cleaning step lacking accuracy and reproducibility, which may affect the cell-type identification process. Therefore it was decided to propose a new and fully-automated method. The batch effect (the technical variance) correction methods are compared in the doctoral dissertation to indicate the most suitable technique and the influence of the correction on the clustering results.

The pipeline is prepared using publically available high-dimensional datasets in order to process Tuberculosis samples provided by the partners.

During the doctoral dissertation, the following theses were formulated:

1) The unsupervised machine learning methods allow data-driven identification of cell subpopulations leading to a better description of the cell-type heterogeneity.
2) Transformation of data feature space into an expanded representation using Gaussian Mixture Modelling allows for better separation and definition of the identified cell populations.
3) A two-step clustering approach with local feature space optimization enables the identification of rare cell populations.

1.3 Chapter contents

The doctoral thesis is organized into four main parts: an introduction to the mass cytometry data analysis explaining the basic terms, data acquisition, and preprocessing steps, and three specific topics that were the subject of detailed studies.

The first chapter concerns the implementation of an automatic solution for the mass cytometry pre-gating step. The second chapter describes the statistical analysis of batch effect removal methods and their influence on clustering results. The last chapter refers to the main goal of the thesis – cell subtypes identification. The third problem was the most examined, occupying most of the doctoral dissertation. The last section includes the results for the Tuberculosis dataset the scientific partners provided.

Each topic is described in detail and includes a brief introduction to the problem, a literature review, a description of the datasets used, a results section, and a discussion section with conclusions. The doctoral thesis is summarized at the end in a few sentences.

1.4 Background

1.4.1 Tuberculosis

Tuberculosis (TB) is caused by a bacteria called *Mycobacterium Tuberculosis*, that in 1882, when Dr. Koch discovered it, killed every seventh person in America and Europe [4]. Today, Tuberculosis is one of the leading causes of death and illness worldwide. Only in 2021, the disease killed over 1.6 million people. The bacteria is spread through the air, making Tuberculosis the second most infectious killer after COVID-19. In 2021 an estimated 10.6 million people fell ill with TB. The disease is present in each country but, when appropriately treated, is curable [5].

Untreated Tuberculosis reaches a death rate of about 50%, but with currently recommended treatment, about 85% of people with Tuberculosis can be cured. The treatment lasts 6 to 12 months and must be strictly followed to prevent the disease's recurrence and bacteria resistance to the drugs.

Not everyone exposed to *Mycobacterium Tuberculosis* will develop the disease. There is also a latent TB infection when the germs are in the organism but are inactive. Such a person does not have TB symptoms and cannot spread the bacteria to others. However, he can develop the disease in the future. Therefore, it is essential to implement preventive treatment. In addition, certain factors increase the likelihood of developing TB, like diabetes, HIV infection, poverty, and smoking. The typical symptoms of Tuberculosis are weakness, cough (also with blood), weight loss, chest pain, fever, and night sweats [5].

The drugs for TB are expensive, especially if the bacteria are resistant to the most effective first-line drugs and sometimes even the second-line ones. The multidrug-resistant TB is a public health crisis; in such cases, the medications are limited, expensive, toxic, and require extensive chemotherapy [4].

Scientists from around the globe are involved in research into therapies, drugs, and ways to prevent the spread of Tuberculosis, but their studies are costly. The studies involve the examination of human cell populations in TB patients by applying the techniques of data science and bioinformatics. A thorough understanding of how infection works, cellular processes, and changes in the body can lead to better ways of dealing with the disease.

1.4.2 Mass cytometry (CyTOF) technology

One of the single-cell technologies is Mass Cytometry by Time-of-Flight (CyTOF). Mass cytometry is an alternative to the older version of the technique, flow cytometry. It overcomes most flow cytometry limitations, like spectra overlap [6], since it uses rare stable isotopes instead of fluorophores to label the antibodies to identify cell types. However, using isotopes also increased dimensionality - from tens to dozens of markers measured simultaneously.

More parameters and millions of measured cells demand higher computational resources and careful analysis. Additionally, because of the difference in technology used for detection and quantification, that is, inductively-coupled plasma instead of photomultiplier tubes, the prepared cells are buried in contrast to flow cytometry where the cells may be sorted and further processed after the signal is recorded.

1.4.3 Sample preparation

Preparing samples for CyTOF involves several steps, but their order may differ depending on the assay type. The first step is the choice of probes (usual antibodies) that will bind and identify specific cell types of interest. The next step is panel design, in which the probes are matched with isotopes. The staining stage allows the conjugation of the probes to ions covalently attached to a metal-chelating polymer. Washing is made multiple times at each step to remove unbound reagents and avoid contamination. Next, each sample cell is covalently stained with a unique combination of barcoding isotopes (e.g., Palladium). The steps allow for running all samples together on the CyTOF, reducing technical variation that may be introduced to the data. After washing, the cells are resuspended in MilliQ containing calibration beads [2], [7].

1.4.4 Mass cytometer

Seven steps of data acquisition from a mass cytometer can be distinguished (Figure 1.1.). The prepared samples with calibration beads are introduced to the mass cytometer, where the nebulizer separates the cells into tiny, single droplets. Each droplet is evaporated and buried at the inductively coupled plasma (ICP) torch at approximately 7,000 Kelvin, which results in the ionization of the cell's metals. The cell is converted into a cloud of ions, but not all of them are of analytical interest. The high-pass ion optic allows filtering out unwanted ions (negatively charged, neutral elements, and positively charged with too small mass). The remaining ions are introduced to the Time-of-flight (TOF) chamber, where they travel with the speed appropriate to their masses – smaller ions travel faster, and heavier ions travel slower. When the ions hit the detector, the signal is recorded as a digitized waveform that is further converted into ion counts. The integrated information for each cell is stored in Flow Cytometry Standard (FCS) file format [2]. Because each sample of cells also contains calibration beads, each observation stored in the file (each row) is called an "event" (cell or bead or contamination).

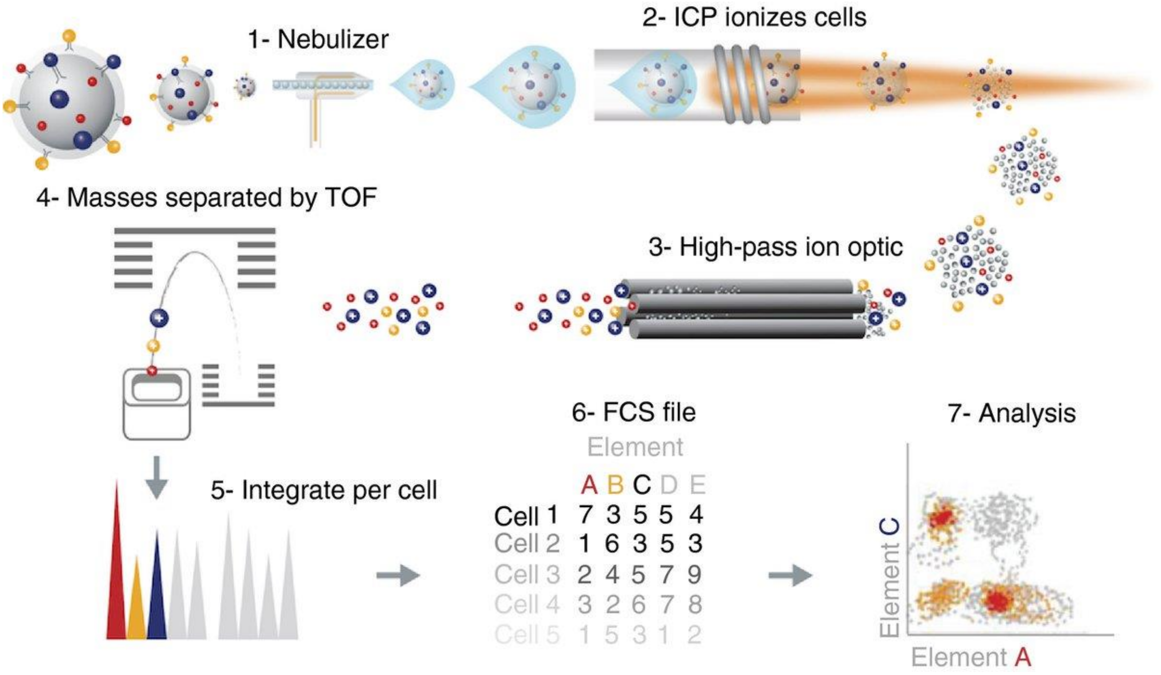


Figure 1.1. Mass cytometer workflow.

1) The cells are labeled with antibodies stained with stable isotopes of rare metals and are separated in the nebulizer. 2) ICP evaporates and ionizes the cells. 3) Resulting cloud of ions is filtered in the high-pass ion optic. 4) Remaining ions enter the time-of-flight detector, where they are separated based on their masses. 5) Each ion hits the detector, and the signal is recorded and integrated for each cell. 6) The information is stored in an FCS file. 7) FCS file is analyzed.

(Source: "Helios, a CyTOF system" by Fluidigm, https://www.fredhutch.org/content/dam/www/shared-resources/fc-cytof/CyTOF-Helios-User-Guide.pdf).

1.4.5 FCS format

Measurements obtained from the mass cytometer are stored in the Flow Cytometry Standard (FCS) file format (Figure 1.2.). The format is a binary data file standard containing the ion count matrix and metadata like experiment information, used markers (antibodies) names, and corresponding isotopes. Usually, for each sample, a separate FCS file is created after the debarcoding step (see *Section 1.4.7. Data preprocessing*) [2].

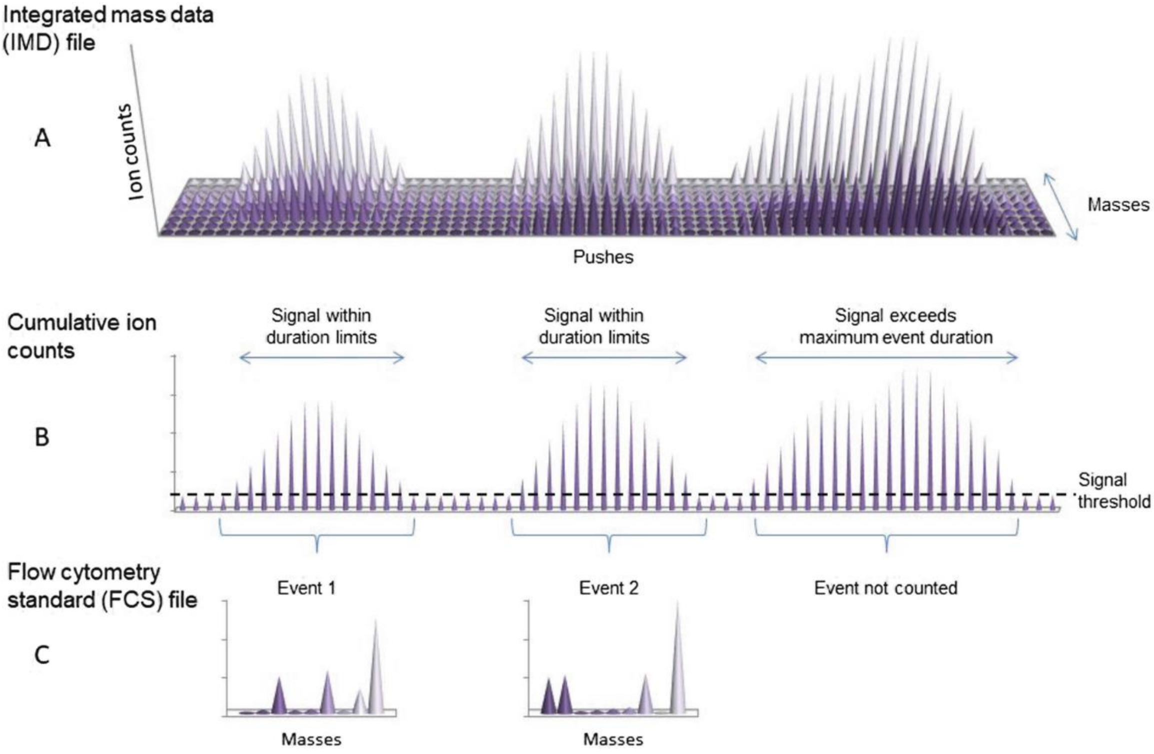
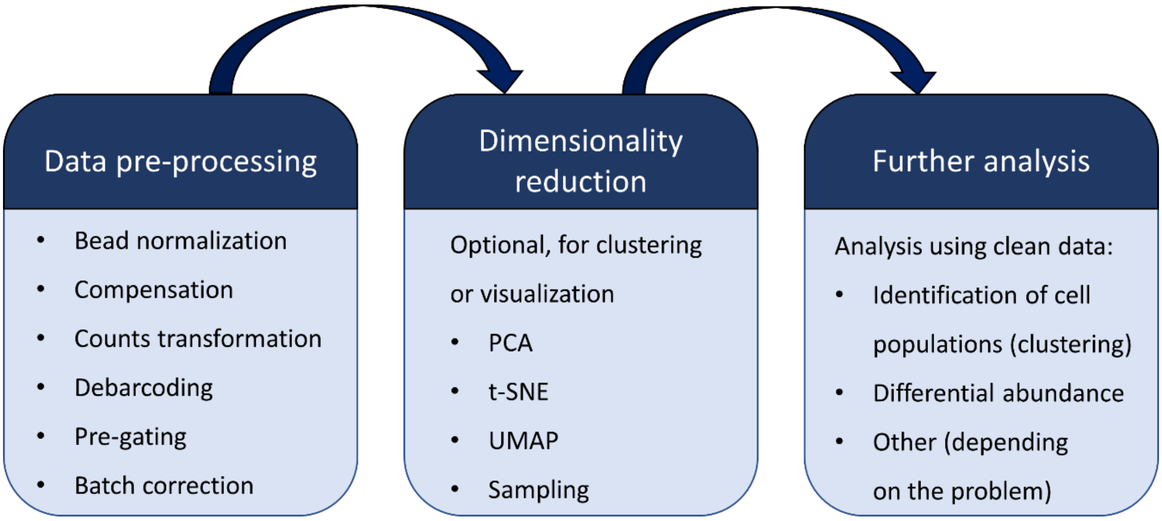

Integrated mass data (IMD) file
A
Ion counts
Masses
Pushes
Cumulative ion counts
B
Signal within duration limits
Signal within duration limits
Signal exceeds maximum event duration
Signal threshold
Event 1
Event 2
Event not counted
Flow cytometry standard (FCS) file
C
Masses
Masses

Figure 1.2. The process of storing ion count information in an FCS file.
(Source: [8]).

1.4.6 Analysis pipeline for mass cytometry data



Data pre-processing

- Bead normalization
- Compensation
- Counts transformation
- Debarcoding
- Pre-gating
- Batch correction

Dimensionality reduction

Optional, for clustering or visualization

- PCA
- t-SNE
- UMAP
- Sampling

Further analysis

Analysis using clean data:

- Identification of cell populations (clustering)
- Differential abundance
- Other (depending on the problem)

Figure 1.3. Main parts of mass cytometry data analysis.
(Source: personal collection).

1.4.7 Data preprocessing

Before any exploratory analysis, the mass cytometry data must be preprocessed to remove the errors and artifacts present after sample preparation in the laboratory. Another critical issue is that the investigation should be performed on single, intact, and live cells. Therefore, dublets and dead cells should be filtered out. For mass cytometry data preprocessing, several different steps can be identified: bead normalization, counts transformation, compensation, sample debarcoding, pre-gating, and batch effect correction.

Bead normalization uses the inserted beads [7] during sample preparation to correct the signal intensity decay and limit the impact of technical variation on the analysis results. The polystyrene beads are used as a reference for the normalization step since the composition of isotopes is known. The two most popular methods can be used for this purpose: the Fluidigm method or MATLAB bead normalization (Figure 1.4.) [9].

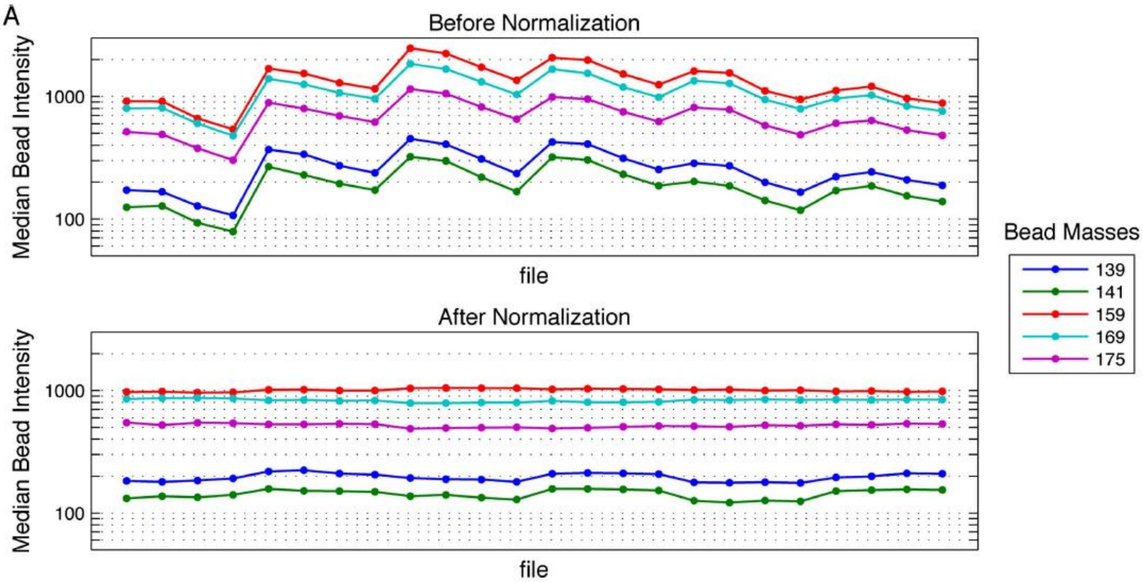


Figure 1.4. Intensity correction with the calibration beads using MATLAB normalizer.
(Source: [9]).

The FCS files contain Di (dual instrument) or Dd (dual data) counts (raw integers or non-zero integers randomized between ranges [x-1, x]. One cell type does not express all of the markers used. Therefore there is a great number of zero values. The distribution of the markers usually is strongly skewed. For better visualization, the ion counts are transformed with ArcSinh or logicle functions (Figure 1.5.). However, the ArcSinh transformation is more often used, with a co-factor of 5 for the mass cytometry data (a co-factor of 150 is intended for flow cytometry) [2].

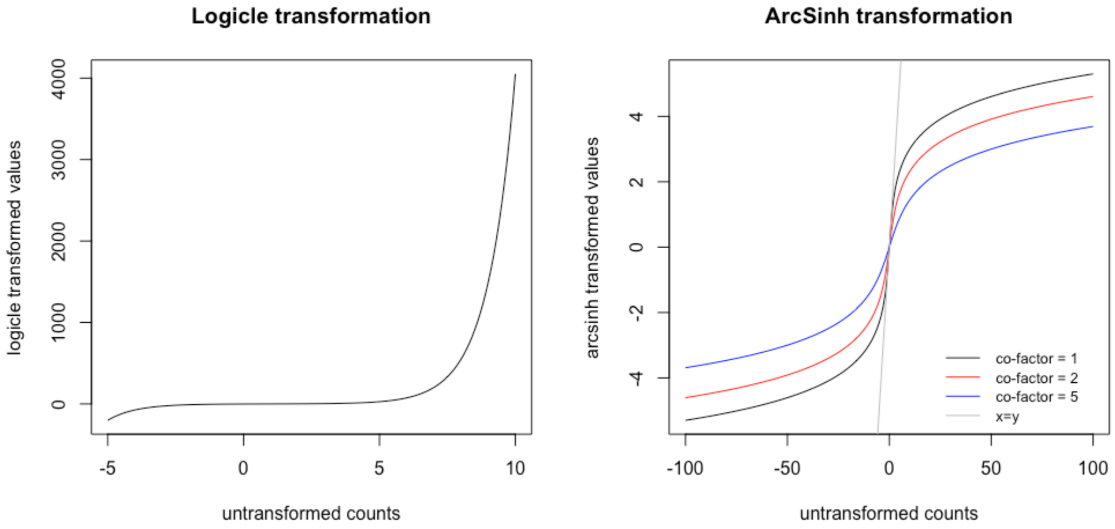


Figure 1.5. Transformation functions for mass cytometry data.
(Source: "Data scientist's primer to analysis of mass cytometry data," The Single Cells Omics Group, https://biosurf.org/cytof_data_scientist.html, access: February 2023)

Compensation, if applicable, corrects so-called "spillover" between the channels, which is usually the problem in flow cytometry cases. Still, studies show that it may also occur in mass cytometry [2].

Different palladium isotope combinations attached to the cells during sample preparation allow running all samples through a mass cytometer simultaneously, decreasing the measurements' technical variation and batch effect. However, during the analysis, it is desirable to separate the events into different FCS files, one per sample (for example, separate cells from other patients). The step is called debarcoding and requires identifying events expressing the appropriate combination of the isotopes [2].

At the beginning of cytometry data analysis, the cell subtypes identification was made manually with a procedure called gating [10], [11]. Some experienced analytics view the channels pairwisely, and on each of the bi-axial plots, they outlined cells of interest, simultaneously excluding dead cells and any doublets. Although it is possible to find cell subtypes in flow cytometry this way, because of the small number of markers, it would be very time-consuming in mass cytometry. Therefore, instead of identifying the subgroups, in mass cytometry data, gating is used only for filtering out debris, dead cells, and doublets (cell-cell, cell-bead, and bead-bead doublets) and is called pre-gating [2], [10]. More about pre-gating can be found in *Chapter 2*.

The last step of data preprocessing is batch effect correction. The batch effect is a technical variation introduced to the data during experimenting, and it makes it challenging to find the natural biological relations [2], [12], [13]. Some tools are for determining the presence of batch effect and removing it from the samples before analyzing. More about the batch effect in mass cytometry data can be found in *Chapter 3*.

1.4.8 Dimensionality reduction and sampling

Because of the high-dimensional nature of single-cell data, the analysis is complicated and requires high computation resources, which are expensive and not always available. One solution to overcome the problem is dimensionality reduction techniques like PCA, t-SNE [14] or UMAP [15]. These techniques allow the application of clustering methods for cell types identification as well as the visualization of the dataset in two/three-dimensional spaces by creating an embedding – low-dimensional representation of high-dimensional information. In other words, these techniques modify the number and presentation of the features (markers, matrix columns). However, the created embedding is less accurate than the original feature space, which may influence the sensitivity of finding rare cell subpopulations. PCA is a linear method that represents linear relationships. Still, it has limitations in terms of variance explained in the first two-three dimensions (for mass cytometry, the first two components capture around 40-50% of variance) that are insufficient to visualize the clusters on two-dimensional plots [10]. In opposition to this, t-SNE and UMAP, as non-linear methods with different approaches for measuring distances, better reflect the relationships between points in 2D space than PCA. However, UMAP projection better reflects the global structure of data than t-SNE, meaning that two clusters of similar points will be placed closer to each other in the UMAP embedding. Still, it is not guaranteed in t-SNE embedding [10], [14], [15].

In mass cytometry, the number of markers (features) is usually around 25-60, which is not as problematic as in, for example, RNA-Seq data, where the number is in thousands.

On the other hand, the number of observations (cells) may reach several million, making it impossible to calculate distances between each pair of cells. Therefore, most of the proposed solutions do not work for so many observations (or have a too long computation time). To overcome the problem, some of them apply downsampling. Sampling (or downsampling) reduces the number of observations, which may cause issues in the case of millions of cells where rare subpopulations may be omitted or less represented in the sampled subspace. As a result, they might be overlooked and never detected. Due to this, in the doctoral dissertation, most of the algorithms do not use sampling if not necessary.

1.4.9 Further analysis

Which major processing steps will be used in mass cytometry analysis depends on the problem and dataset. The clean dataset can be used for further investigation, like cell-type identification or differential abundance. The doctoral thesis focuses on finding cell populations and all the necessary steps that have to be taken in order to prepare the data for clustering.

1.4.10 Clustering for the cell-type identification task

Cluster analysis is a machine learning technique that tries to find structures in data in an unsupervised way – without annotations provided by the user. Usually, the task is unsupervised if the labels are unavailable and hard to create. Clustering (or grouping) is more complicated than classification tasks, where the model learns patterns in data with prior knowledge about the expected outcome [16]. The clustering algorithm defines a set of rules to group observations with similar characteristics. The perfect result is when each cluster contains similar observations and is dissimilar to others [17], which means the groups are homogeneous. Several families of grouping techniques can be distinguished: hierarchical, partitional, probabilistic, and density-based [16].

The most popular and fundamental, but influential in its simplicity, partitional clustering algorithm is k-Means. The idea behind k-Means is that each cluster can be represented by a centroid, a point in the N-dimensional feature space created from the mean feature values of the cluster members. The algorithm assigns each point in space to the closest, based on Euclidean distance, initial centroids updated at each iteration with the help of an objective function. The model training stops when there is no significant improvement in the position of the centroids. However, the algorithm has disadvantages – it requires the user to specify the number of groups that must be found.

Moreover, the created clusters have regular shapes that may not suit the data structure and lead to erroneous results. Finally, and most importantly, despite being one of the fastest algorithms compared to others, it does not scale well to substantial volume datasets since it iterates over all data points. Therefore, generating results may take a lot of time [16].

The issue of the circular clusters can be fixed with Gaussian Mixture Model (GMM), which assumes that the data points are Gaussian distributed. The two parameters: mean and standard deviation values, determine the shape of the groups. GMM, as the name indicates, contains from two to multiple Gaussians (components) in one or N-dimensional space. The observations are assigned to the components based on probability – the closer the point is to the particular Gaussian center, the more likely it is to belong to that cluster. The randomly initialized parameters are updated with an optimization algorithm called Expectation-Maximization (EM). However, GMM has more parameters than k-Means that have to be tuned. Moreover, searching for the optimal number of components and their parameters for a high-dimensional dataset may be time-consuming [18].

Another extensively used and simple technique is agglomerative clustering. In this bottom-up approach, each data point is considered a single cluster. The clusters are merged together based on the closest distance until one cluster containing all the data is formed [18]. The technique may also work as a top-bottom version, i.e., hierarchical clustering, that starts treating all points as one cluster and successively divides them into smaller ones. The results can be presented as a dendrogram – a tree diagram following the clustering steps. The algorithms do not need the number of final clusters since it can be chosen as the best-split step. The methods are suitable when the data is hierarchical; however, they suffer from low efficiency.

Other types of popular algorithms are, among others, DBSCAN, Affinity Propagation, MeanShift, and BIRCH and their modifications. In addition, traditional machine learning techniques are often used as a base algorithm for the more complex and advanced methods [16], [18].

1.5 Materials

In the doctoral thesis, several mass cytometry datasets were used. The Tuberculosis dataset was obtained from scientific partners from Stellenbosch University, RPA, and four other datasets were publically available. Depending on the provided information by each dataset, they were used for developing different algorithms. The public datasets are described in detail in the doctoral thesis topics where they were used.

1.5.1 Tuberculosis dataset from Stellenbosch University

The dataset consisted of cells obtained from 21 patients belonging to three groups: drug-resistant Tuberculosis (TB), Other Lung Diseases (OLD), and Healthy Donors (HD). Other Lung Diseases included subjects with diseases like HIV, Hilar Lymphadenitis, Bronchiectasis, SIADH, Asthma, and cancer and were recruited from the Pulmonology Division of Tygerberg Academic Hospital. Healthy subjects were recruited from regions with high TB infection pressure. The cells came from bronchoalveolar lavage fluid (BALF) obtained through bronchoscopy in Tygerberg Hospital (TBH). Samples were prepared in three simulation conditions: unstimulated (UNS), stimulated with Tuberculin purified protein derivative (PPD) and with Phytohaemagglutinin (PHA). The dataset was measured in seven batches with the CyTOF2 instrument in the SATVI institution (South African Tuberculosis Vaccine Initiative) at the University of Cape Town [19]. The details of the gathered samples can be found in Table 1.1. The experts manually gated the dataset to remove debris, dead cells, and doublets.

Table 1.1. Details about the Tuberculosis dataset from Stellenbosch University.

					Simulation conditions		
Group of patients	No. of patients	Age range	Sex (M/F)	Smoking status (Y/N)	UNS	PPD	PHA
TB	7	33-53	6/1	7/0	7	7	2
OLD	7	27-78	1/6	3/4	7	7	2
HD	7	20-69	2/5	5/2	7	7	2

In total, there were 10,065,441 cells. The cells were stained with an antibody panel containing 32 targets. Nineteen of the markers were extracellular, describing phenotypes of the cells, and 13 were intracellular, telling the cell's functions. The panel is described in Supplementary Table 6.1.

1.5.2 Public datasets

Four high-dimensional publically available datasets were used during the doctoral studies. Three of the datasets were used for the pre-gating approach development and one for the cell-type identification process. Table 1.2. summarizes the datasets.

Table 1.2. Summary of the public datasets used in the doctoral thesis.

Dataset	No. of events	No. of markers	Annotated	Description	Purpose in the doctoral thesis
Leipold and Maecker (2015) [20]	5,250,000	33	No	21 peripheral mononuclear cell (PBMC) samples	Pre-gating
Trussart *et al.* (2020) [12]	8,817,845	31	No	24 peripheral mononuclear cell (PBMC) samples from patients with chronic lymphocytic leukemia (CLL) and healthy donors	Pre-gating
Simoni *et al.* (2017) [21]	2,523,240	38	No	Human innate lymphoid cells (ILC)	Pre-gating
Samusik *et al.* (2016) [22]	514,386	38	Yes	Ten mouse samples with annotations for 24 cell types	Cell-type identification

1.6 UMAP and mISO plots for visualization of high-dimensional data

Because of the high dimensionality of the datasets, it is a challenge to visualize the results of clustering or other characteristics in an easy-to-interpret way. Therefore, the UMAP dimensionality reduction technique was used to create a low-dimensional representation for visualization purposes. UMAP in each case was created using the Python package 'umap-learn', with the following parameters: n_neighbors=30, min_dist=0.2, metric='euclidean' or 'hamming' (for binary data).

However, indicating clusters with different colors on the UMAP plot when the number of observations (cells) is over a million and they overlap in the embedding often results in meaningless visualization. To overcome the problem, a visualization technique called median isoline (mISO) plot was proposed [19], which can be superimposed on UMAP plots. The density of points is determined by isolines using parameter m (defaults to 0.5), which indicates the density level above which the data will be shown. mISO plot presents the regions of the highest concentration within a group of points, for example, clusters or samples (Figure 1.6.).

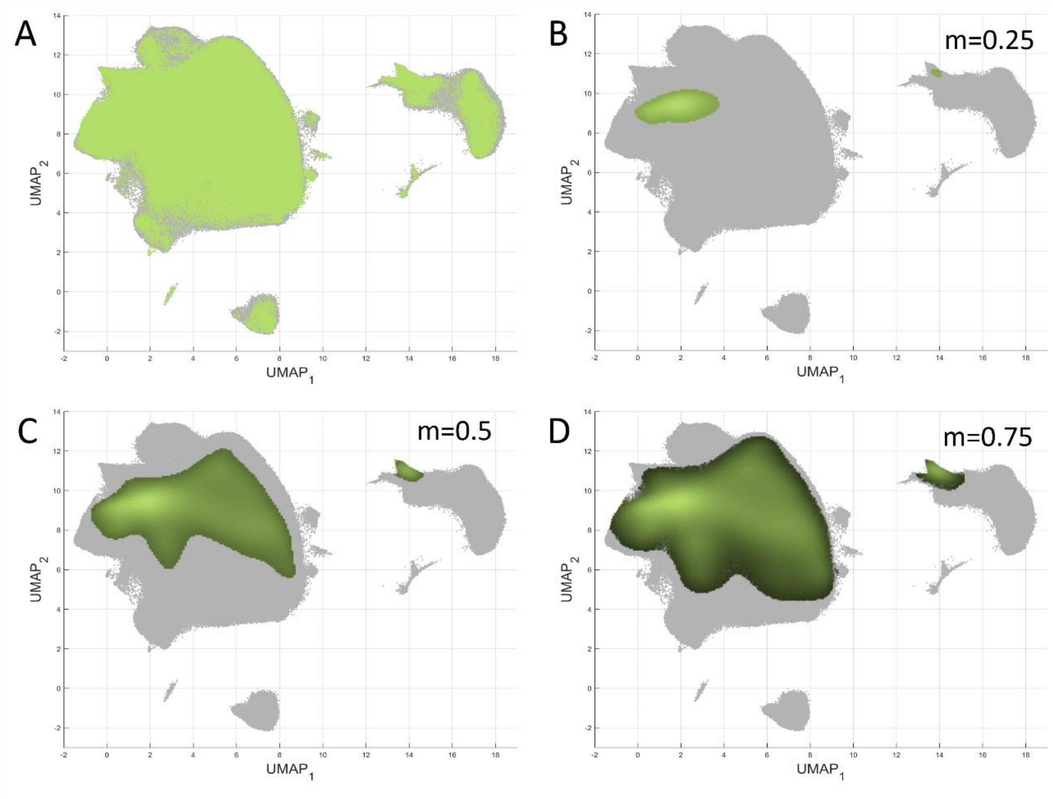


Figure 1.6. Exemplary of mISO plots – proposition of high dimensionality data visualization method.
A) The scatterplot of a dataset in the UMAP space. Green points represent observations belonging to one cluster, and gray points are other observations in the dataset. B) mISO plot with m parameter set to 0.25 showing a region in the UMAP space above 25% quartile isoline. C) mISO plot showing region above 50% quartile isoline. D) mISO plot showing region above 75% quartile isoline. (Source: [19]).

2 Pre-gating of mass cytometry data

2.1 Introduction to pre-gating

In mass cytometry, pre-gating allows for the removal of debris, dead cells, and double events. Firstly, pre-gating was done manually, making the process ineffective and prone to errors. Also, the results were non-reproducible. Recently, with more automated solutions, the user still has to set some parameters that may influence the results, like a gate shape [23]. Moreover, the proposed solutions are not appropriate for high-dimensional datasets with millions of events. Therefore, as a part of the doctoral thesis, we proposed a new, fully-automated solution to this problem. The analysis details and results were published in [24].

Some ready-to-use gating methods exist, like OpenCyto [25] or FlowCal [26]. However, they were developed primarily for cell subtypes identification in flow cytometry data, and they do not perform well on mass cytometry data [27]. On the other hand, GateFinder [28], Cluster-To-Gate [29] and HyperGate [30] are mass cytometry methods intended for gating hierarchy generation given target cells to provide a better explanation of results for non-technical experts like biologists. Another problem with the existing algorithms is that they often use downsampling, which may lead to the omission of rare cell populations.

In [24], we proposed a method for mass cytometry data pre-gating that is based on the Gaussian Mixture Model and clustering of its components. Because the technique is automated, the results are reproducible. Moreover, it is independent of the sample size, so appropriate for high-dimensional datasets.

2.2 Materials and methods

Five different markers are used in the pre-gating of mass cytometry data: DNA1 (isotope Ir191Di), DNA2 (Ir193Di), Bead (Ce140Di), Dead (In115Di), and Event length. The event length is a duration of a typical cell ion cloud measured in 200-300 microseconds [31]. With the markers, four two-dimensional visualizations can be created:

- DNA1-DNA2 - to distinguish cellular events from debris and cell-cell doublets,

- Bead-DNA1 – to identify polystyrene beads added to the samples for normalization purposes,
- Dead-DNA1 - to identify dead cells,
- Event length-DNA1 - to identify doublets (cell-cell, cell-bead) and beads.

However, not always all of the markers are available. Three publically available datasets (Table 1.2.) were used for the experiments since the Tuberculosis dataset from Stellenbosch University partners was received after the manual pre-gating by the experts. The number of available pre-gating markers varied between the datasets and is summarized in Table 2.1. Because the dataset from Leipold and Maecker (2015) had all the required pre-gating markers (complete information), it was used as a training set – the proposed algorithms were developed based on the dataset. The remaining two datasets were used to validate the implemented algorithms. A visualization of the bi-axial plots for each of the data is presented in Figure 2.1. The datasets and calibration beads were normalized and transformed with the ArcSinh function.

Table 2.1. Details on the dataset used for the implementation of the pre-gating method.
(Source: [24])

Dataset	Pre-gating markers	Bi-axial plots (gates)
Leipold and Maecker (2015)	DNA1, DNA2, Bead, Dead, Event length	DNA1-DNA2, Bead-DNA1, Dead-DNA1, Event length-DNA1
Trussart *et al.* (2020)	DNA1, DNA2, Event length	DNA1-DNA2, Event length-DNA1
Simoni *et al.* (2017)	DNA1, DNA2, Dead, Event length	DNA1-DNA2, Dead-DNA1, Event length-DNA1

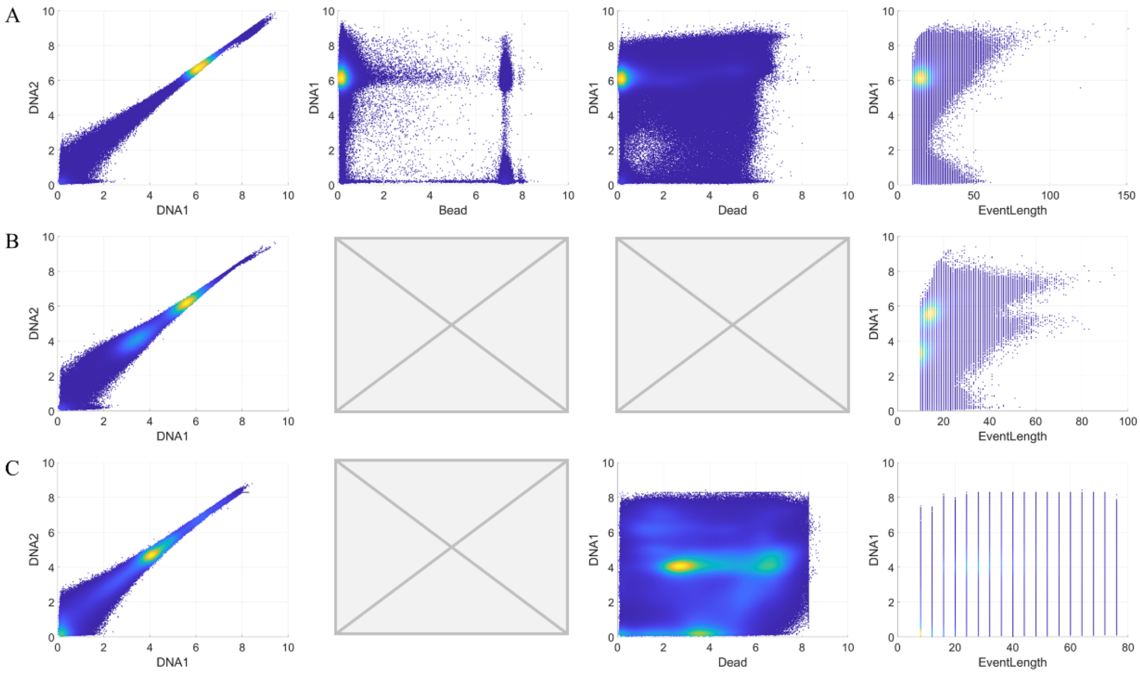


Figure 2.1. Bi-axial plots were created with pre-gating markers for filtration purposes.
The observations are colored by density estimate, where yellow indicates regions of the highest concentration. The first column shows DNA1-DNA2 channels bi-axial plot; the second column – Bead-DNA1; the third column – Dead-DNA1 and the last column – Event_length-DNA1. A) Leipold and Maecker dataset. B) Trussart *et al.* dataset. C) Simoni *et al.* dataset. (Source: [24]).

The proposed method is fully automated. Firstly, two-dimensional Gaussian Mixture Model (GMM) decomposition [32] was performed on each bi-axial plot, and the resulting components were clustered. The Expectation-Maximization algorithm and Bayes Factor [33] as the stopping criterion found the optimal number of components. Usually, bigger aggregates of cells are described as a mixture of Gaussian components. Therefore, they lie close to each other in the feature space. Grouping the elements allows for finding the aggregates and creating final clusters of events. The next step calculates a conditional probability for each observation to assign it to the correct group. The filtration of unwanted events is based on the final clusters.

2.2.1 Clustering algorithm

Components resulting from 2D GMM can be visualized as rotated ellipses in a 2D space where the center of each ellipse is the mean value represented as a point (x,y), and a covariance matrix determines the shape. Because of the high number of observations, the densest areas on the bi-axial plots result in many components describing one aggregate of cells that should form one cluster. These aggregates are the most important for pre-gating since they contain live, intact, single cells.

The proposed solution searches for so-called big seeds - the component mean values. The component proportions are sorted in descending order and added successively until 90% is reached. The rest of the components are called small seeds.

$$\mathrm{d} = \frac{|\text{vector length}|}{\sqrt{\mathrm{s}^2\text{pooled}}} \tag{2.1}$$

$$\mathrm{s}^2\text{pooled} = \frac{\mathrm{a}_1\mathrm{s}_1{}^2+\mathrm{a}_2\mathrm{s}_2{}^2}{\mathrm{a}_1+\mathrm{a}_2} \tag{2.2}$$

Between all the seeds, a distance measure is calculated. The distance measure is an effect size (2.1) defined as the length of a vector between two seeds divided by their pooled variance (2.2), where s_1^2, s_2^2 are new variances determined by the intersection of the ellipses with the line (vector) connecting the seeds. And α_1 and α_2 are components' proportions. Using 1D GMM on the distribution of the distances, the lowest cut-off threshold was determined as an intersection of two first conditional probability lines. The point defines the maximum distance needed to merge two big seeds or two small into one cluster. In the opposite case, the seed creates a new cluster. Then, the rest of the small seeds are joined to the created clusters.

The result of the above step is a smaller number of groups of events. The conditional probability lines determine the assignment of each event to the appropriate cluster – the event belongs to the cluster for which the probability is the highest.

2.2.2 Pre-gating filtration criterion

If an event belongs to at least one cluster that contains less than 15% of data is discarded from the analysis. The criterion was established after examining the percentage of data in the training set clusters knowing which clusters probably contain unwanted events.

2.3 Results

The 2D GMM was fitted to each pair of markers, constituting a 2D plot used for pre-gating, and the resulting components were aggregated into final clusters. The number of components varied between 13 and 15, and they were joined to create three to five clusters depending on the feature space. A distance threshold was found for each pair of the pre-gating markers based on the 1D GMM decomposition of the calculated effect sizes. The details of the resulting number of components and clusters and distance thresholds are summarised in Table 2.2.

Table 2.2. Pre-gating results for Leipold and Maecker's dataset.
(Source: [24])

Pre-gating plot	No. of components	Distance threshold	No. of clusters
DNA1-DNA2	15	4.82	3
Bead-DNA1	13	5.64	5
Dead-DNA1	15	6.97	4
Event length-DNA1	14	5.49	4

Each event was assigned to the appropriate cluster by the highest conditional probability. If an event was assigned to at least one cluster containing less than 15% of observations, it was discarded from the analysis. The whole process constitutes the proposed automatic pre-gating approach. Results for the training dataset (Leipold and Maecker) are presented in Figure 2.2. After the automatic pre-gating, the observations decreased from 5,250,000 to 4,170,565 live, intact, and single cells ready to be analyzed (red dots in Figure 2.2., third column).

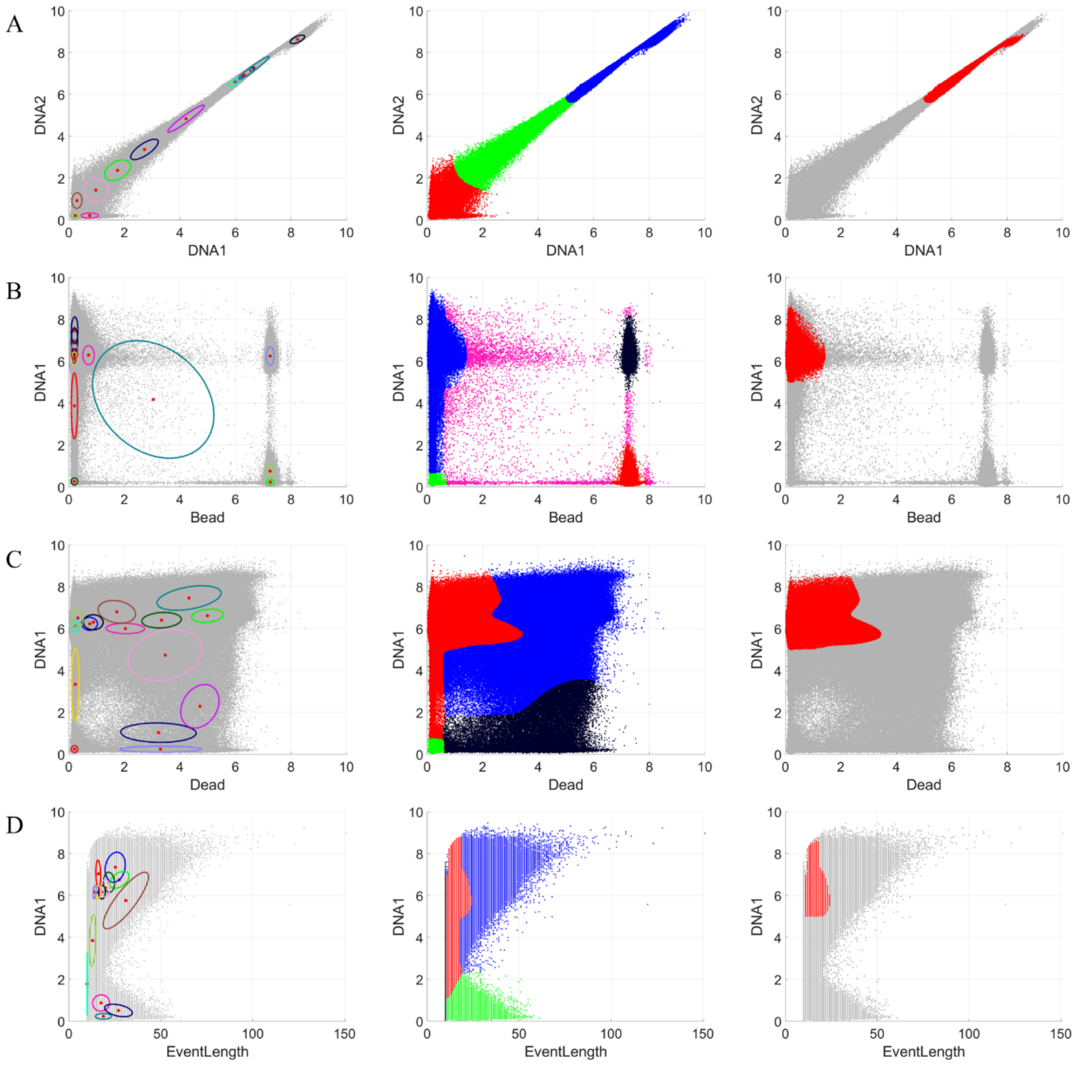


Figure 2.2. Visualization of the results for Leipold and Maecker's dataset.
The first column shows the resulting 2D GMM components; the second column shows observations assigned to the appropriate cluster based on the highest conditional probability; the third column – the final result of pre-gating, red points indicating live, intact, single cells that remain in the further analysis. A) DNA1-DNA2 markers with 15 components organized into three clusters. B) Bead-DNA1 markers with 13 components organized into five clusters. C) Dead-DNA1 markers with 15 components organized into four clusters. D) Event_length-DNA1 markers with 14 components organized in four clusters. (Source: [24]).

The training set's thresholds were applied to the validation datasets: Trussart *et al.* and Simoni *et al.* to check the generalization abilities. The results are presented in Figure 2.3. Details on the number of filtered events are contained in Table 2.3.

Table 2.3. Pre-gating results. The number of removed and remaining events for each dataset. (Source: [24])

Dataset	No. of events	No. of events after pre-gating	No. of removed events
Leipold and Maecker	5,250,000	4,170,565	1,079,435
Trussart *et al.*	8,817,845	8,059,990	757,855
Simoni *et al.*	2,523,240	1,666,255	856,985

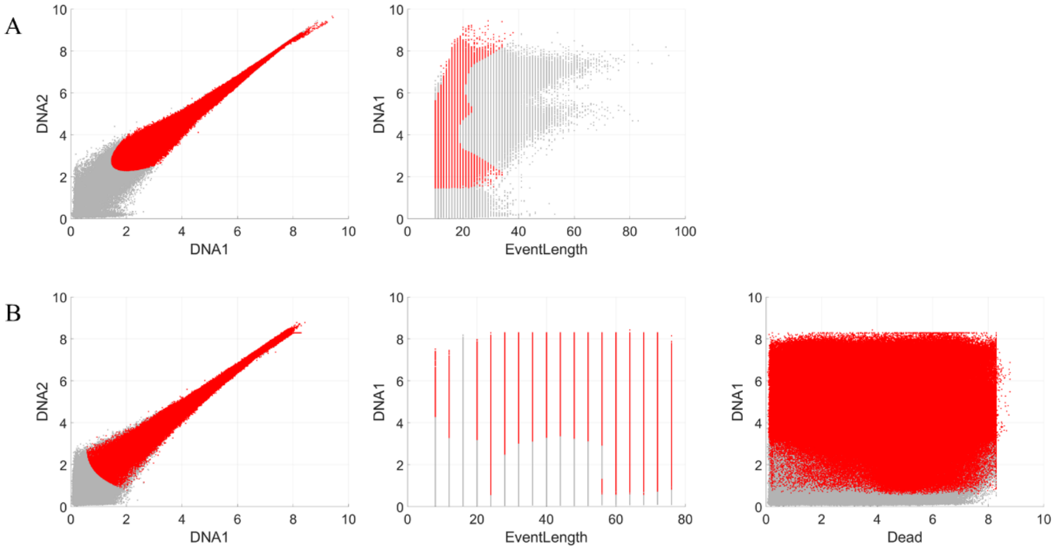


Figure 2.3. Pre-gating results for validation datasets.
The pre-gating was conducted using available pre-gating markers. A) Trussart's dataset. B) Simoni's dataset. (Source: [24]).

2.4 Discussion and conclusions

In the study, a new and fully automated method for pre-gating for mass cytometry was proposed that works for high-dimensional datasets without downsampling, which may lead to removing rare cell subpopulations. In the doctoral dissertation, a rare subpopulation is considered a small group containing less than 1% of all dataset cells. The algorithm makes the results reproducible, which is impossible for manual pre-gating. Moreover, all parameters are data-driven. However, the algorithm may work incorrectly if the dataset contains more debris than intact cells because it depends on the density of observations in the two-dimensional space.

Results for the training set are satisfactory – the proposed clustering algorithm merged the GMM components into bigger aggregates based on the density of events. It resulted in clusters with either a large or a small number of observations. This step enabled defining the filtration criterion based on the percentage of the dataset contained in the clusters. It is known where areas on the two-dimensional plots of markers have unwanted events; for example, the DNA1-DNA2 plot (Figure 2.2.A) bottom-left corner indicates debris. After applying the proposed approach, it can be observed that this region is included in one cluster containing approximately 5% of the data points. These events are probably assigned to other unwanted clusters on other bi-axial plots. In the same graph, the top-right corner usually contains cell-cell doublets. Although there is no separate cluster for the area, those events are removed (Figure 2.2.A) with the help of the other cluster assignments on different pairs of markers' space.

After examining the resulting clusters in each 2D feature space, the filtration criterion was defined as a threshold of at least 15% of dataset observations that must be contained in a cluster. Otherwise, the cluster of events is discarded. This threshold may be too high or too low in some cases, but the results for validation datasets show that it was suitable for them to get good results (Figure 2.3.).

It can be concluded that the filtration criterion removes too many observations from Leipold and Maecker's dataset, more than needed to be removed. On the other hand, a lower filtering threshold resulted in the inaccurate removal of known unwanted observations. Some of the cell-cell doublets and debris were left in the dataset. As mentioned before, the particular observation is discarded from the analysis if it belongs to at least one cluster that contains less than 15% of the data. It will be investigated in future work if the condition should be modified to observations that belong to at least two or more such clusters, which could preserve more live, intact cells.

The implemented clustering method gives satisfactory results when dealing with different data distributions in the two-dimensional space of pre-gating markers (Figure 2.2.). Cluster shapes are other than those defined in [11] since the method is based on the density of points. This makes the gates more accurate as the regular shapes may include more events that should be removed.

The proposed approach does not require all five pre-gating markers, which is its advantage since some of the markers may not always be available, like in the validation datasets. For example, datasets from Trussart *et al*. and Simoni *et al*. have different markers available, but the results are still promising. However, with the smaller number of bi-axial plots, the number of conditions is also smaller, and the result may be less accurate – fewer events may be discarded from the analysis (Table 2.3.).

The existing solution presented in [33] is also based on GMM. However, the method is intended to identify cell subtypes in flow cytometry. The first step is finding dead cells with channels for forward and sideward scatter light not present in mass cytometry data. Then they cluster the data to find the populations of cells. The proposed approach in the doctoral thesis aims to filter out unwanted events rather than find cell populations. It enables the data clean-up without user intervention since all the parameters are data-driven.

One drawback of the presented method is the computation time for finding the optimal number of GMM components with the Expectation Maximization algorithm when the dataset reaches millions of events. It took approximately 12-48 hours, depending on the dataset. This is a trade-off between computation time and the accuracy of the pre-gating by discarding the possibility of using downsampling. The computation time should be optimized in future work. The calculations were done using MATLAB 2020a on GeCONiI servers (AMD CPU 256 threads, 2.6G, 2TB RAM).

In summary, the proposed method's results are satisfactory for high-dimensional data. The pre-gating approach provides complete automation and, therefore, reproduction of the results. It can be generalized to other datasets that contain fewer pre-gating markers removing unwanted observations from the analysis. The solution does not apply downsampling, preventing the removal of rare cell subpopulations. However, it can be extended with additional filtration criteria to make the results more reliable and optimized to reduce the computation time.

3 Batch effect correction analysis

3.1 Introduction

One of the critical steps in mass cytometry data analysis is batch effect correction [2]. The batch effect is a technical variance introduced to the data, making it difficult to find real biological relationships. Therefore, much effort is put into implementing methods that effectively remove technological artifacts. The batch effect can result from experimental design, but it also occurs between datasets acquired from different experiments or technologies (like single-cell RNA-Seq). It is possible to merge data from different experimental techniques to gain more in-depth knowledge about the biological problem, and the batch effect makes it almost impossible [13].

A few methods for batch effect removal are dedicated to mass cytometry data, like CytofBatchAdjust, CytoNorm, CytofRUV, iMUBAC, and cyCombine. The first one, CytofBatchAdjust [34] requires technical replicates to be included in each run that works as a reference to which the samples are adjusted without manual intervention. Similarly to CytofBatch Adjust, CytoNorm [35] depends on an identical control sample to be included in each batch to apply the correction between sets. The algorithm uses FlowSOM [36] to find cell subpopulations. For each cluster, quantiles are computed to determine goal distributions to which the original values are translated, and then samples are normalized to remove technical variation. CytofRUV [12] is based on the RUV-III method that was dedicated to technologies like TNA-Seq or microarrays. It requires so-called pseudoreplicates to estimate the variation introduced to the data. iMUBAC [37] uses only healthy (control) samples for batch effect correction. Firstly, the dataset is downsampled to a fixed number of observations per batch, and then the sets are corrected with Harmony [38] with the default parameters. The most recent method, cyCombine [13] is based on ComBat [39], which was initially implemented for batch effect reduction in microarrays to remove the batch effect from cell clusters resulting after using a self-organizing map. The method allows the integration of data from different batches, experiments, and also different experimental techniques.

The mentioned approaches also offer a lot of diagnostic plots and measurements that indicate the presence of batch effect and the effectiveness of the applied correction. Examples of such diagnostic measures are Earth Mover's Distance (EMD) which compares batches on different types of plots, the comparison of EMD values before and after correction [40], and the marker's distribution in each batch. Other examples are multidimensional scaling plots (MDS) and visualizations of the batches in 2D space after dimensionality reduction. However, the mentioned techniques, especially the last one, are not accurate indicators of the batch effect when millions of cells are given.

To examine the effectiveness of batch effect correction methods for high-dimensional data, they were decided to be applied to the subset of the Tuberculosis dataset obtained from Stellenbosch University (*Chapter 1, Section 1.5.1.*). Since the data does not contain technical nor biological replicates that are required by three of the mentioned methods (CytofBatchAdjust, CytoNorm, CytofRUV), only two methods were compared: iMUBAC and cyCombine. Because each process works differently, there is a chance that the corrections change expression values significantly, which will affect cell population identification, like the number of found cell types. Therefore, PARC [40] algorithm was used to determine the influence of batch effect correcting methods on identified cell groups. Furthermore, since iMUBAC only uses healthy samples for correction, cyCombine was also applied to the same subset of data.

The analysis and results were described in [19] and presented during the international conference IWBBIO 2022, Gran Canaria, Spain.

3.2 Materials and methods

3.2.1 Dataset

Data used in the analysis is a subset of the Tuberculosis dataset from Stellenbosch University, restricted only to the healthy samples of bronchoalveolar lavage cells (BALC). The dataset was normalized using MATLAB Normalizer v0.3 software, and the samples were manually pre-gated by experts to exclude debris, dead cells, and beads. The expression values were ArcSinh transformed with a co-factor of 5. The total number of cells belonging to the healthy subjects was 4,145,712. Table 3.1. presents details about the used dataset subset.

Table 3.1. Number of cells in each batch from the subset of the dataset obtained from Stellenbosch University.
(Source: [19]).

Batch number	Number of cells in the batch
Batch 1	761,230
Batch 2	598,492
Batch 3	205,958
Batch 4	329,228
Batch 5	341,007
Batch 6	1,449,084
Batch 7	460,713
Total	**4,145,712**

3.2.2 Batch effect correction

The dataset was corrected with iMUBAC using most of the parameters with default values except for the parameter "maxN" which was set to 300,000 (larger than the default value). This parameter is used for downsampling the dataset – a maximum of 300,000 observations were randomly selected from each batch for the batch effect correction. The downsampling resulted in a total of 2,005,958 cells, approximately 50% of the data. For a fair comparison, the same observations were applied to cyCombine correction even though the method has no restrictions on the dimensionality of the dataset and the patient's status. cyCombine was also used with default parameters [19].

3.2.3 Cell subtypes identification after batch effect correction

After the batch effect correction, cell populations were identified with PARC [40] algorithm with default parameters. The method was chosen because of its scalability, satisfactory results reported, and consistency with the experts' knowledge and because the method does not require specifying the prior number of clusters. PARC constructs a nearest-neighbor graph with a hierarchical navigable small world, and based on the distribution of weights, it prunes the graph's edges. The Leiden algorithm is used to find communities that create the cell populations. The algorithm works fast and accurately for high-dimensional data, effectively finding rare cell subpopulations.

The clustering algorithm was applied to the expression values of 32 markers after the batch correction, and the resulting assignments were transferred to the corresponding uncorrected observations for comparison purposes [19].

3.2.4 Statistical comparison of batch effect correction methods

With expression values before and after the batch effect correction and cell populations found by PARC, the effect of iMUBAC and cyCombine on clustering results was examined for each marker with the ANOVA post-hoc Q Tukey test. The expression values of markers between clusters were compared using an effect size measure (3.1) where mA and mB are mean values of given marker expression in clusters A and B; N_{ps} is a pooled sample size (3.2); N_A and N_B are the numbers of observations belonging to clusters A and B; N is a total number of cells; SS_{within} is a sum of squares within k clusters.

$$d_{AB} = \frac{mA-mB}{SE*\sqrt{N_{ps}}} = \frac{mA-mB}{\sqrt{\frac{SS_{within}}{N-k}*\frac{1}{N_{ps}}}*\sqrt{N_{ps}}} = \frac{mA-mB}{\sqrt{\frac{SS_{within}}{N-k}}} \tag{3.1}$$

$$N_{ps} = \frac{2}{\frac{1}{N_A}+\frac{1}{N_B}} \tag{3.2}$$

The result of the pairwise comparison was a set of d_{AB} values for each marker. Each marker's global effect size measure was calculated as a median value from the collected pairwise measurements. Therefore, for each technique (iMUBAC, cyCombine), 64 values in total were obtained – 32 values of global effect size before the correction and 32 values after the modification. The global effect sizes were then compared with the Wilcovon signed-rank test with the assumption that after the correction, the clusters returned by PARC should be better separated, therefore, have a higher heterogeneity of marker expression between clusters than before the modification. For the test, a significance level of 5% was assumed. With the p-value, a median shift of values was also provided.

Additionally, centroids of clusters after iMUBAC and cyCmbine batch effect correction were grouped with an agglomerative clustering algorithm with Spearman's rank correction coefficient as the distance metric. The goal was to find pairs of equivalent cell types in both approaches. If the cluster's expression profile from one method matches the cluster's expression profile from the second method – the two groups probably contain the same cell population [19].

3.2.5 UMAP transformation

For the analysis, the Uniform Manifold Approximation and Projection (UMAP) [15] method was used to visualize the results. As mentioned in *Section 1.4.2.2*, UMAP is one of the dimensionality reduction techniques used mainly for projecting high-dimensional data on two-dimensional space, making the results more interpretable. The study generated UMAP embedding for samples after batch effect correction, and then the same transformation was applied to raw data. This technique was tested on different data types and presented in [41]. The assumption was that each cell would move from its position (x_1, y_1) in the UMAP space before correction to the new place (x_2, y_2) after the correction, which is caused by the change of expression values. Thanks to that, the correction effect can be observed. A simple regression neural network consisting of three fully-connected layers was proposed to apply the same UMAP transformation to new data points. Each layer's neurons were 100, 50, and 25. ReLU was used as an activation function. The network maps the 32 markers' expression values into the learned UMAP embedding described by two values. The performance of the network was evaluated with the coefficient of determination [19].

3.2.6 Technical details

The comparison was conducted with three programming environments: Python, R, and MATLAB 2020a. The calculations were carried out using the GeCONiI server (Intel Xeon Gold 6226R CPU 64 threads, 2.9GHz, GPU: 3x NVIDIA Tesla V100-PCIE with 1x 16GB and 2x 32GB).

3.3 Results

Using default parameters, the same subset of cells from the Tuberculosis dataset provided by Stellenbosch University was applied to batch effect correction with iMUBAC and cyCombine. The result of the batch effect correction was presented on mISO plots in the UMAP space. The effect of iMUBAC modification is visualized in Figure 3.1. It can be noticed that the uncorrected samples are located in different regions on the mISO plot (Figure 3.1.A), while the corrected samples partially overlap (Figure 3.1.B). The samples show the same behavior before and after cyCombine correction (Figure 3.2.). Despite differences in the point locations that result from separate UMAP model (the same parameters but the corrected values differ), each sample occupies a different space before the correction (Figure 3.2.A) and share the area with other samples after the modification (Figure 3.2.B). It can also be observed that the overlapping is more significant and more regular for cyCombine than iMUBAC correction.

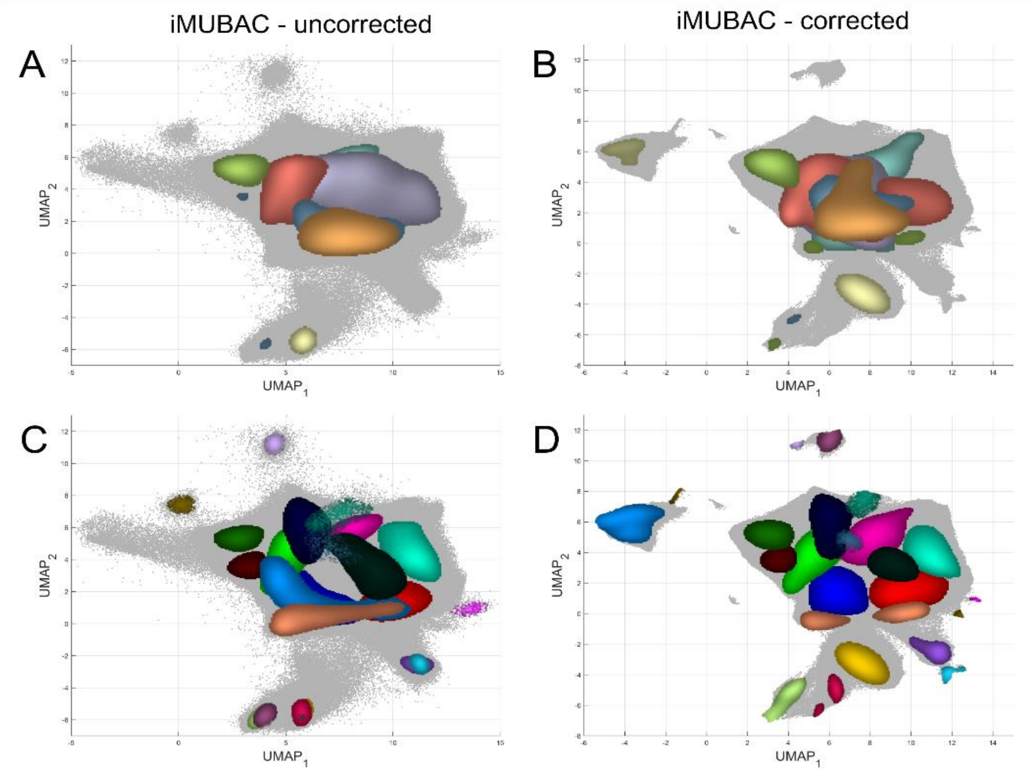


Figure 3.1. mISO plots before and after iMUBAC correction.
Each color indicates one sample (one batch) of data (A-B) or one cluster (C-D). A) Samples before batch effect correction. B) Samples after batch effect correction. C) Visualization of clusters found by the PARC algorithm on the raw data. D) Visualization of clusters found by the PARC algorithm on the dataset with corrected expression values. (Source: [19]).

The PARC clustering algorithm was supposed to find cell types in the dataset after each correction. Then the assignments of observations to appropriate clusters were transferred to observations before the modification (raw expressions). The results are also presented on the mISO plots to expose the effect of correction on each cell population placement in the UMAP space. After iMUBAC correction, 22 clusters were found (Figure 3.1.D) that overlapped when transferred to data before correction (Figure 3.1.C). After cyCombine correction, 18 groups were found (Figure 3.2.C-D) with a similar impression to iMUBAC but weaker.

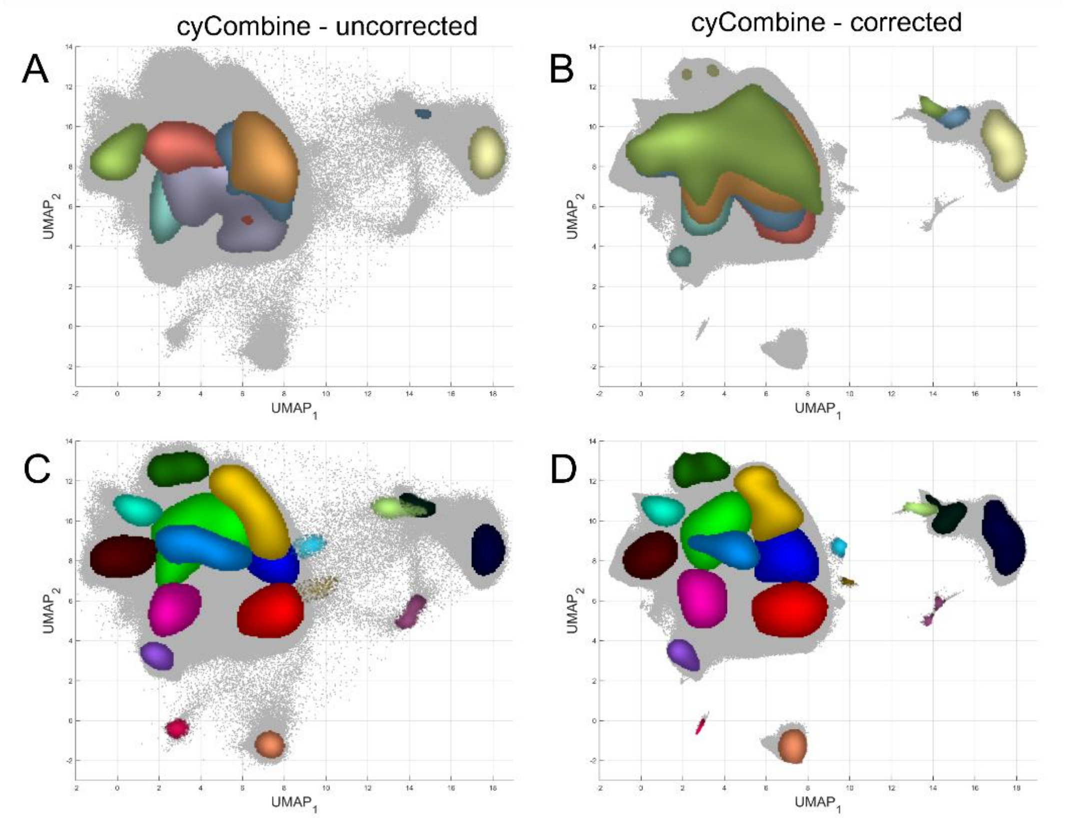


Figure 3.2. mISO plots before and after cyCombine correction.
Each color indicates one sample (one batch) of data (A-B) or one cluster (C-D). A) Samples before batch effect correction. B) Samples after batch effect correction. C) Visualization of clusters found by the PARC algorithm on the raw data. D) Visualization of clusters found by the PARC algorithm on the dataset with corrected expression values. (Source: [19]).

The PARC's cluster centroids were subjected to agglomerative clustering and presented as a dendrogram (Figure 3.3). The goal was to find similar clusters (containing probably the same cell type) between the two approaches. As seen on the plot, most of the discovered groups (14 exactly) from one experiment have a pair of similar cells in the other experiment. The pairs are indicated with the same color on the dendrogram and mISO plots (Figure 3.3). Clusters that do not have a similar pair are presented in gray.

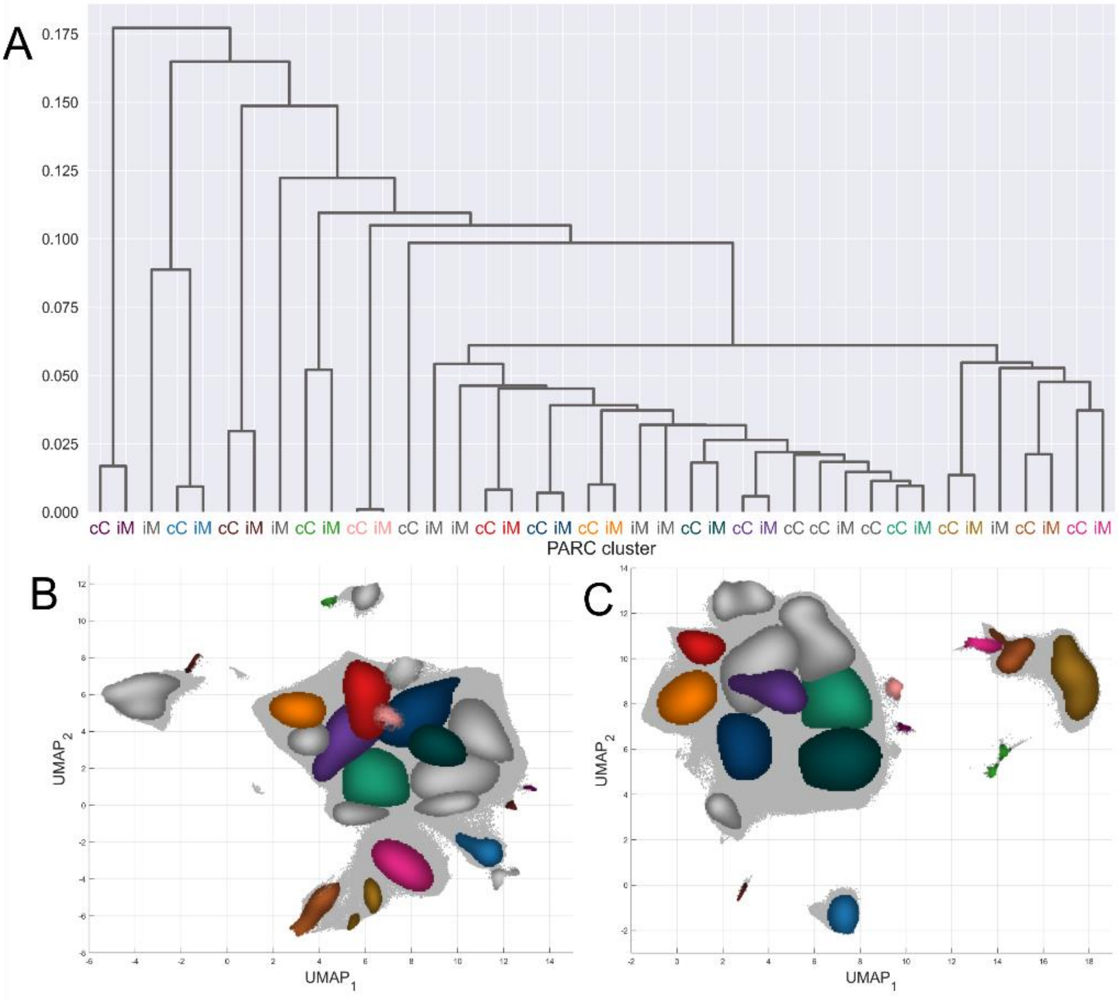


Figure 3.3. Comparison of the PARC clustering results after batch effect correction.
Similar clusters between the experiments (iMUBAC and cyCombine correction) share the same color. However, if the cluster is shown in gray, it has no matching pair among the other experiment result. A) Dendrogram after clustering of the clusters' centroids. iM – centroid of a cluster created after iMUBAC batch effect correction; cC – centroid of a cluster created after cyCombine batch effect correction. B) Result of PARC clustering on data after iMUBAC correction. C) Result of PARC clustering on data after cyCombine correction. (Source: [19]).

To further examine the differences between the found clusters after each correction method, marker expression values were compared among the groups using the ANOVA post-hoc Q Tukey test and the calculated effect size measures. Figure 3.4. presents the median d_{AB} values for each marker before and after iMUBAC and cyCombine corrections. After applying the iMUBAC correction to the data, the effect size values changed slightly compared to those calculated before the modification. On the other hand, after using the cyCombine correction, the values increased noticeably. The Wilcoxon test verified the observation. For the iMUBAC approach, the p-value was equal to 0.4628, and the median shifted to 0.0011. For the cyCombine method, the p-value and mean shift were 4.38e10^{-7} and 0.0961, respectively.

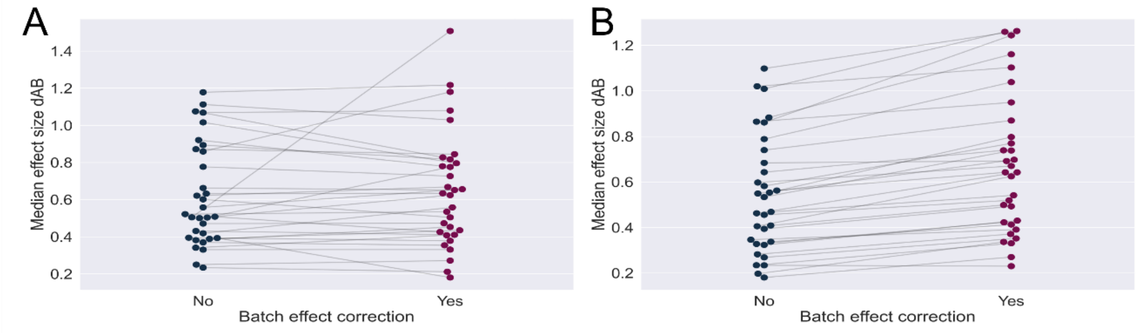


Figure 3.4. Comparison of the median effect sizes from post-hoc ANOVA test.
Each graph shows median effect size values before and after batch effect correction with A) iMUAC and B) cyCombine. (Source: [19]).

3.4 Discussion and conclusions

Several methods for batch effect correction in mass cytometry data have been proposed, but the most popular ones are CytofBatchAdjust, CytoNorm, CytofRUV, iMUBAC, and cyCombine. The first three algorithms require technical or biological replicates to be included in the dataset, but it is not always possible. Someone may ask which of the method to choose and how they impact the expression values and, consequently, the clustering results. To address this question, a comparative analysis has been conducted that examines the differences between groups of cells identified after batch effect correction with the use of two of the methods: iMUBAC and cyCombine. The same subset of cells from Healthy Donors from the Stellenbosch University dataset and default parameters were used for each algorithm to make a fair comparison.

Marker expression values after correction with each method were used to generate UMAP embedding showing the two-dimensional representation of the data. The learned transformation was then applied to raw expression values (before the correction) using a simple neural network for the regression task. Since UMAP takes a vector (one observation) of values and maps it into coordinates in the 2D space, the resulting coordinates will change if the values are changed a bit. Therefore, observing the initial coordinates of the cells before the correction concerning the positions after the modification was possible. The effect of the transformation can be noticed in Figure 3.1. and Figure 3.2. as some “blurring” occurring on the plots on the left in the panels. When a batch effect is present in the data, it is assumed that each sample (batch) will take a different place in the UMAP space because of the technical variation that distinguishes them. Therefore, the samples should overlap after the correction since the artificial differences are reduced. This behavior can be observed in Figure 3.1.A-B and Figure 3.2.A-B.

The mISO plots show that the iMUBAC correction had less impact on the expression values than cyCombine, which is reflected in the points’ positions in the UMAP space. In the cyCombine case, the samples overlap almost entirely after the correction. Therefore, it can be concluded that cyCombine offers better correction than iMUBAC.

PARC clustering algorithm with default parameters was applied to the corrected marker expression values to find cell populations. The algorithm works effectively for high-dimensional data and automatically determines the optimal number of clusters. After the iMUBAC and cyCombine correction, the method found 22 and 18 clusters, respectively. The cluster assignments were also used with the uncorrected expression values to visualize the difference in the placement of the cell groups. The results are presented in Figure 3.1.C-D and Figure 3.2.C-D. As seen in the visualizations, the clusters before expression correction overlap. Without the batch effect correction, PARC applied directly to the raw data would instead join the groups together rather than find the separate clusters. As a result, some of the cell types could not be discovered. It confirms the need for a good batch effect removal technique. It is expected that after the correction, the clusters will be separated since the technical variation will be reduced, and the biological variation will influence the points’ position in the UMAP space in a way that similar cells from all samples will lay close to each other.

Visualization alone is not sufficient to determine which of the methods is better. However, it can be interesting to find out which set of clusters that were found by the PARC algorithm is better. The dendrogram (Figure 3.3.A) presents that 14 of the groups are similar between the approaches. Moreover, there are similarities between the layout of the identified cell populations of each of the methods – for example, the blue cluster lay close to the purple group on both mISO plots despite different UMAP transformations (Figure 3.3.B-C). Therefore, the clusters of the same color probably contain the same cell types. On the other hand, eight iMUBAC and four cyCombine clusters do not have a pair of similar cell types.

The cluster labels were used for the expression values before the correction to examine how each cluster's marker expression values changed after the batch effect correction. The effect size was calculated in pairwise comparisons between found cell populations for each experiment and marker. The median value of each marker's effect size measures was considered as a global effect size for that marker. It resulted in 32 effect size measures before and after the correction (64 values per experiment). The Wilcoxon signed-rank test was used to determine if there were significant differences in the effect size measures. For iMUBAC correction, the difference was not significant with a p-value of 0.4628 which may indicate that the batch effect was not reduced effectively and the identified clusters may contain a mixture of different cell types. It was also checked how many clusters result from clustering uncorrected data directly, and PARC found 24 cell populations. After the iMUBAC correction, the number decreased to 22, a relatively small reduction compared to cyCombine (18 clusters). This may indicate that the iMUBAC correction is insufficient to remove the batch effect from mass cytometry data.

Based on the mentioned observations and results, it can be concluded that cyCombine is a better method for batch effect correction in mass cytometry data. It is not limited to healthy samples and does not use downsampling – it works efficiently for high-dimensional datasets. Also, after the correction, the homogeneity is significantly higher than before the modification (p-value=$4.38e10^{-7}$).

Nevertheless, the topic should be further analyzed and compared with other methods for removing technical variance on different datasets. For example, the cyCombine technique may be better for a certain dataset, not general.

In summary, in the study, two batch effect methods were compared – iMUBAC and cyCombine, to examine their effectiveness in reducing technical variance introduced artificially to the data and the impact of the correction on the clustering results. The cell type marker heterogeneity increased after cyCombine correction in contrast to the iMUBAC correction that did not end in significant improvement. The results indicate the superiority of cyCombine over iMUBAC for the Tuberculosis dataset from Stellenbosch University limited to the Healthy Donor samples [19].

4 Identification of cell subpopulations

4.1 Introduction

There are many clustering methods dedicated to cell-type identification in mass cytometry data. Most of the solutions adapt classical machine learning clustering techniques and some data preprocessing, like, for example, hierarchical clustering combined with density-based downsampling of the observations. Currently, the best methods [3] in terms of detection sensitivity, stability of the results, and time of computation are FlowSOM [36] and PhenoGraph [42]. Another group of solutions is those dedicated to different biological datasets, including mass cytometry, like PARC [40] or ClusterX [43]. A summary of the existing methods is presented in Table 4.1.

SPADE [44] contains four main modules: data downsampling, clustering, creating a minimal spanning tree to connect the clusters, and upsampling, where the remaining cells are mapped into the found clusters. The method requires four input parameters: markers used to build the tree – which relies on the knowledge of which features may be helpful; the density of outliers and targets, which controls the downsampling process; and the number of clusters. Another solution, DensVM [45] applies t-SNE dimensionality reduction, and the resulting embedding is fed to the clustering and 2D peak-finding algorithms. Each peak represents a cluster centroid to which cells are assigned based on the smallest distance. Cluster assignments obtained from the clustering algorithm are used as labels in the Support Vector Machine (SVM) model for classification. Cells the method fails to label are used as a test set in SVM training.

Another algorithm is based on k-Nearest-Neighbor (k-NN): ClusterX [43] which works with datasets with up to five features. If the dataset has many markers, ClusterX reduces the dimensionality with t-SNE. For each point in the space, two measurements are calculated: local density and the distance to points with higher local density. That way, the centroids are determined, and the rest of the points are assigned to the closest one creating a cluster. FlowSOM [36] is a method that is based on Self-Organizing Map (SOM). It can analyze flow and mass cytometry data in a two-level clustering. The algorithm builds and trains SOM; the resulting nodes are connected in a minimal spanning tree. The last step is meta-clustering with a known number of clusters. Alternatively, an "elbow" method can be applied to find the optimal value. However, the authors advise setting the number of clusters as more than expected.

In PhenoGraph [42] authors used a nearest-neighbor graph (k-NNG) where cells are represented by nodes connected to other most similar cells. In the next step, community detection is performed to find phenotypically similar cell subpopulations. immunoClust [46] uses Finite Mixture Models to cluster cell events in the first step, and then the cell clusters are grouped and merged across samples (meta clustering) to get the final results. X-Shift [22] is an algorithm based on k-NN density estimation. For each point, a density estimate is calculated, and the points are connected with the closest nearest neighbor that has a higher density estimate. The point is considered a potential cluster centroid if no such neighbor exists. In the end, some centroids are connected, and the resulting clusters are merged iteratively based on Mahalanobis distance until a specified condition is reached.

PAC-MAN [47] divides the data space into smaller hyper-rectangles based on density. Then it uses k-Means to round and merge the clusters. The method analyses each sample separately and combines the results. PARC [40] constructs a nearest-neighbor graph with a hierarchical navigable small world, and based on the distribution of weights, it prunes the graph's edges. The Leiden algorithm is used to find communities that create the cell populations. The algorithm works fast and accurately for high-dimensional data, effectively finding rare cell subpopulations.

4.1.1 Limitations of the existing methods

Although many existing methods exist for identifying cell populations, none is perfect. The first limitation is the number of observations that can be processed. Some of the algorithms are not scalable. They calculate pairwise distances between the observations, resulting in a huge matrix that can not fit in memory, or they use different computationally expensive transformations. They may work well for datasets up to a few hundred thousand cells, but most of the time, they are not able to process millions of cells.

Table 4.1. Summary of the existing methods for cell population identification in mass cytometry data.

Method	Base algorithm	Number of observations	Downsampling	Needs the number of clusters
SPADE, 2011 [44]	Agglomerative clustering	Thousands	Yes	Yes
DensVM, 2014 [45]	SVM	Thousands	No	No
ClusterX, 2014 [43]	k-NN	Thousands	No	No
FlowSOM, 2015 [36]	SOM	Millions	No	Yes
PhenoGraph, 2015 [42]	k-NNG	Millions	Yes	No
immunoClust, 2015 [46]	FMM	Thousands	Yes	No
X-Shift, 2016 [22]	k-NN	Millions	No	No
PAC-MAN, 2017 [47]	k-Means	Millions	No	Yes
PARC, 2020 [40]	k-NNG	Millions	No	No

The second limitation is downsampling. Some methods, like SPADE, reduce the number of cells in order to apply an algorithm that does not scale for high-dimensional datasets. As a result, some rare cell populations may never be detected, mainly if the dataset contains millions of cells and a specific rare population has only dozens of cells. The stochastic nature of downsampling in the SPADE algorithm also makes the results non-reproducible, leading to different outcomes between iterations [48].

Moreover, many of the proposed visualizations of results and the clustering techniques are meaningless regarding cluster positions relative to each other, which may lead to wrong assumptions [48].

And finally, the knowledge about the expected number of cell populations in the data. It is hard to estimate how many cell populations should be expected, particularly taking into account the heterogeneity of data. The cells may be in a different cell cycle stage, switch between states, and therefore exist in a continuum of development, which means that cell types may overlap [2]. Experts try to annotate mass cytometry data through manual gating, plotting pairs of markers against each other, and highlighting regions of interest. This is done using markers that are known to be related to specific cell types. However, this is only a subset of all possible 2D bi-axial plots that can be created based on the markers. Some relationships between other features and the cell types may not be well understood or discovered. It is almost not possible to check all the possibilities; for example, if 38 markers describe a set of cells, it will result in 706 bi-axial plots of each marker-vs-marker. It would require a lot of hours of working or a team of experts, which also introduces bias to the result. Therefore the manual gating is usually made by a few experts (for the comparison and finding the consensus) on a known subset of the combinations, and the annotations may be less accurate. This problem is highlighted by many authors [2], [44], [46] who point out that relying on expert annotations to assess the quality of clustering approaches may not be the best idea.

4.2 Materials and methods

4.2.1 Experiments

The doctoral thesis presents a set of experiments that explore the ideas described in the section below. They include calculations on regular and expanded feature domains and the appropriateness of using subpopulations. All the analyses leading to the implementation of the final pipeline for mass cytometry cell-type identification were conducted with the use of Samusik's dataset:

- Comparison of existing solutions in the regular domain – this also includes the comparison of model evaluation metrics before and after taking into account the cell subpopulations,
- Impact of the expanded feature space on the existing solutions and comparison of the results with the regular domain,
- Comparison of different dataset transformations in the expanded domain,
- Comparison of the various dataset transformations in the expanded domain after applying feature selection,

- The proposition of a divisive approach.

The final approach with the parameters and criteria established with Samusik's dataset was then applied to the Tuberculosis mass cytometry dataset provided by partners from Stellenbosch University. Because the dataset is unsupervised (in contrast to Samusik's dataset, the cell assignment is unknown), the results were presented in a slightly different form, with additional statistical tests and plots.

4.2.2 Characteristics of Samusik's dataset

Because the Tuberculosis dataset provided by partners from Stellenbosch University is not annotated, evaluating different methods for cell subtypes identification using the dataset would be challenging. Therefore, a publicly available dataset from Samusik *et al.* [22] was used to compare and develop different algorithms.

As it was presented in Table 1.2., the dataset consists of 514,386 cells categorized into 24 types. The cells were annotated by three experts in the manual gating process. The cells come from 10 mouse samples labeled with 38 markers. The dataset was preprocessed and corrected for batch effect.

To better understand the dataset, UMAP-based mISO visualization was generated (Figure 4.1.). Appropriate cell-type names were written near each cluster on the visualization.

As can be noticed in Figure 4.1., in the UMAP space, cell types biologically similar are placed closer than the rest of the cell groups. To further investigate the UMAP space, Figure 4.2. presents the distribution of expression values of exemplary markers. The lighter the color (yellow), the higher the marker value for the cells (represented as points) in that UMAP area.

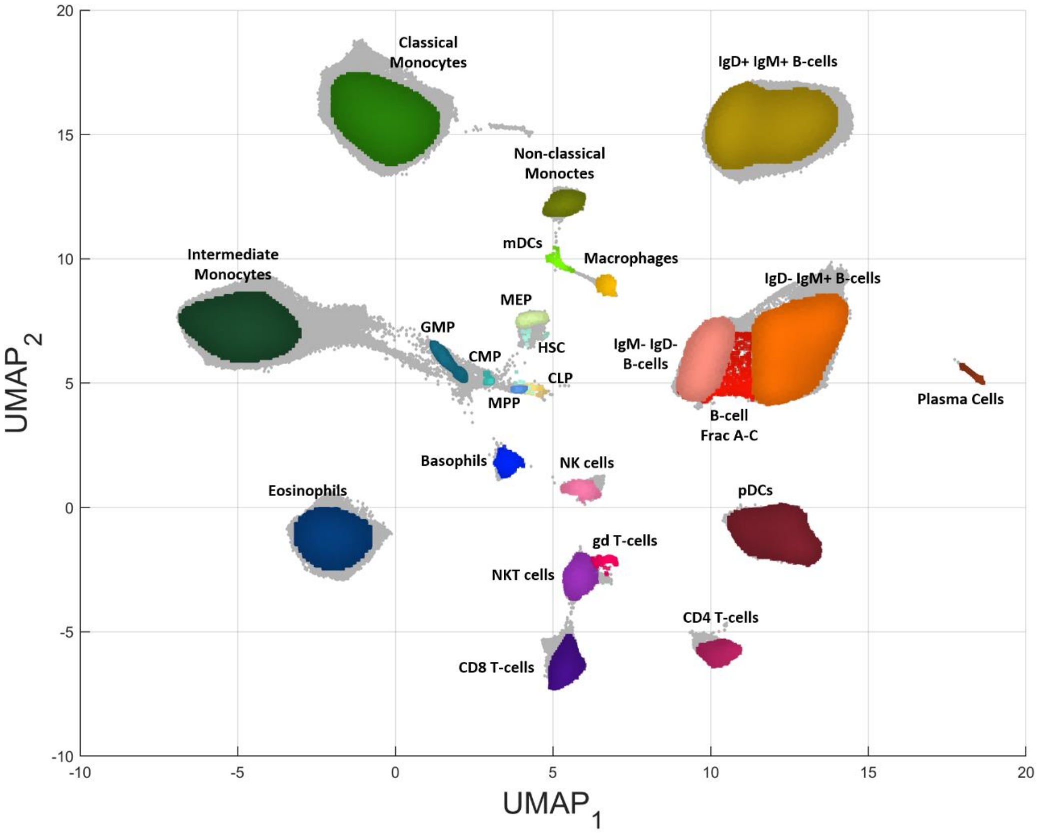


Figure 4.1. mISO plot for Samusik's dataset.
Each color indicates a different cell type whose name was provided next to its cluster. (Source: personal collection).

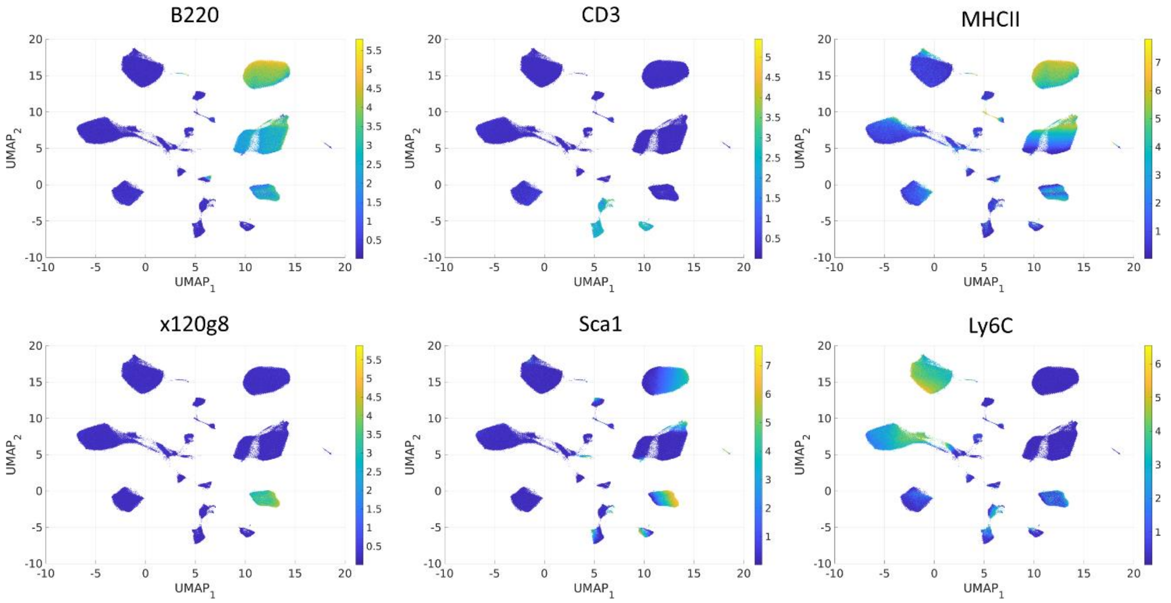


Figure 4.2. Distribution of exemplary marker values in the form of a heatmap.
The color indicates the expression of the marker from dark blue (regions of no/weak expression) to yellow (high expression). (Source: personal collection).

Figure 4.2. shows that even within one cell type, there are differences in marker expression. For example, some pDCs cells have a deficient expression of the Sca1 marker, while some are characterized by their high expression. These nuances result from the data heterogeneity since the cells may be in different states [2]. Heterogeneity may significantly impact the clustering results – clustering algorithms may find more groups of phenotypically similar cells than the number identified by an expert. It is visible on a three-dimensional plot describing the density of points in the UMAP space (Figure 4.3.C).

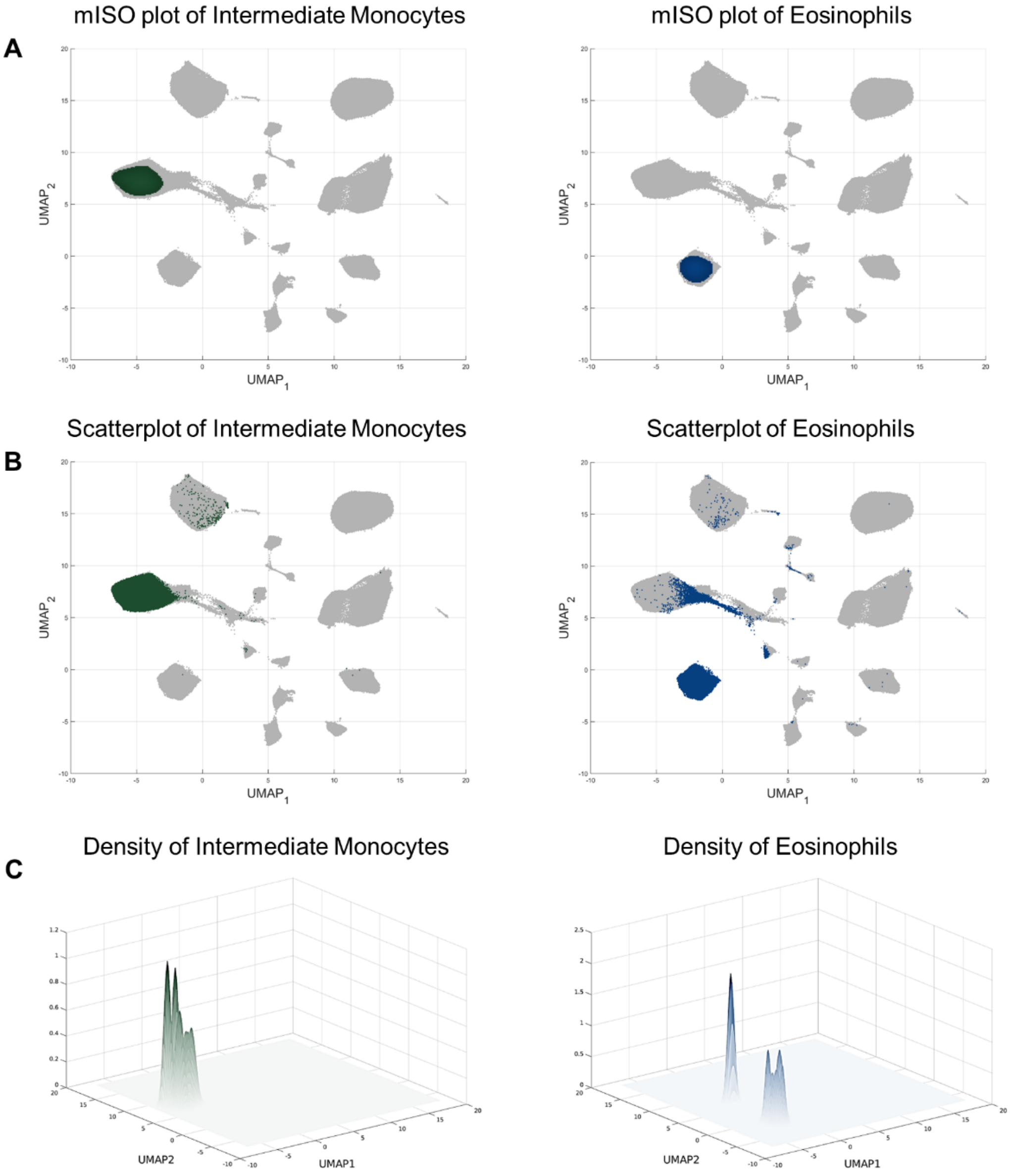


Figure 4.3. Distribution of cells within two exemplary cell types caused by inter-cluster heterogeneity. The grey color shows all cells in the dataset; colored points belong to the specific cell type. A) mISO plots for cell type. mISO plot shows the area on the UMAP 2D plane that is the most densely occupied by the cells. B) Scatterplot presenting all the cells that belong to the group. C) 3D density plot – each peak represents a potentially different subpopulation of the cell type. (Source: personal collection).

Without heterogeneity, it is expected to observe only one distinct peak since the point should occupy the area almost uniformly. If there are more well-separated peaks, it can be assumed that the differences in their marker profiles are high enough to identify the cell-type subpopulations. That means the probability that the subpopulations will be assigned to one cluster is low, and the identification process will result in overestimating the number of groups.

4.2.3 Data heterogeneity

Each cell type was separately applied to Gaussian Mixture Model decomposition in two-dimensional space to examine the data heterogeneity and how many subpopulations a clustering algorithm may find in Samusik's dataset. The result of 2D GMM Is a set of parameters for each found subpopulation: mean value as a point (x, y), which also determines the centroid of a subpopulation, component proportion, and covariance matrix. The optimal number of components included in the Gaussian Mixture Model was determined with Bayes Factor [49]. As can be observed in Figure 4.3.B, some cells may be only outliers instead of a group of cells that somehow behaves differently than the rest of the cells of their type. They may overlap with another cell type, resulting in inaccurate manual gating (*Chapter 2*).

Therefore, it was decided to verify which of the resulting 2D GMM components should be considered true subpopulations of a given cell type. The component proportion parameters were collected for each cell type, and the cut-off threshold was found using the one-dimensional GMM. Components with a proportion higher than the threshold were considered potential subpopulations. The final number of the likely subpopulations was the maximum expected number of clusters created from the cell type.

Given the set of defined cell subpopulations, the identified cell-type clusters were assigned to the closest one. The assignment was done in the marker feature space – for each group, a cluster centroid was computed and compared with the centroids of the found subpopulations. Then, the cluster received a new label indicating the identified cell type, and the evaluation metrics were calculated to score how good is the division.

4.2.4 The idea behind the expanded feature space

The biggest challenge in mass cytometry cell-type identification problems is identifying rare populations. The provided markers create an almost unique marker profile for each of the types, depending on their low/high expression. While the large cell populations with easily distinguishable marker profiles are relatively easy to identify, the rare ones may have too weak marker profiles to be noticed. For example, they may express marker values in the middle of the marker's range values, making the information harder to extract. Thus the idea of expanded feature space was considered.

The idea has its roots in the assumption that if a particular cell's profile is somehow hidden by another cell's profile, it can be extracted through the decomposition of the marker values. Since the marker's distribution is a mixture of Gaussians, each Gaussian potentially describes a different set of observations. Each marker value may be assigned to the appropriate Gaussian Mixture Model component by calculating the conditional probability – the cells that express a specific marker value belong to the component with the highest conditional probability. It also belongs to other components with smaller probabilities. Consequently, the calculated conditional probabilities create new features with values bonded to a range [0, 1]. It was assumed that some components contain information primarily for a rare cell population in that expanded feature space. The idea is visualized in Figure 4.4. What can be noticed, the distribution of the features changes drastically.

The Gaussian Mixture Model for each marker was found using the Expectation Maximization algorithm and Bayesian Information Criterion [50] for selecting the optimal number of components to obtain the expanded feature space. Conditional probability was computed for each marker's GMM. The expression values were translated to the conditional probabilities for each cell to create new features. However, the conditional probability may express some artifacts that must be corrected before using them as a new set of features. The probabilities were converted with the approach described in *Section 4.2.5*. The final new features after correction represent a membership function.

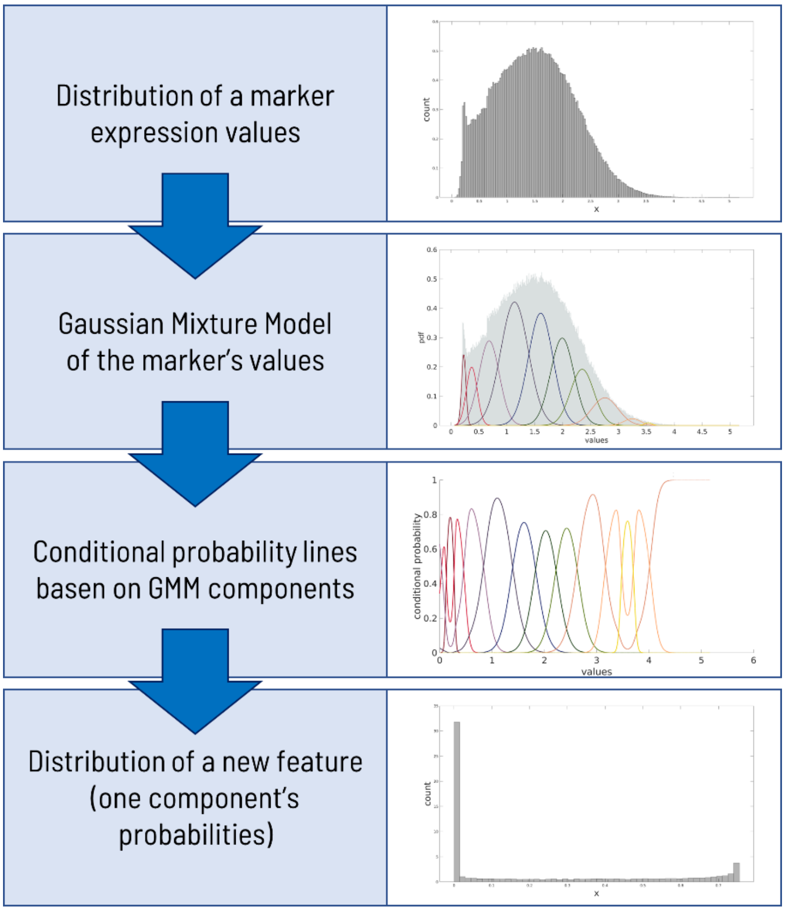

Distribution of a marker expression values
count
x
Gaussian Mixture Model of the marker's values
pdf
values
0
0.1
0.2
0.3
0.4
0.5
0.6
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
Conditional probability lines basen on GMM components
conditional probability
values
0
0.2
0.4
0.6
0.8
1
2
3
4
5
6
Distribution of a new feature (one component's probabilities)
count
x

Figure 4.4. The idea behind the expanded feature space.
The data distribution (each marker) is decomposed with GMM into a set of components that possibly describe different groups of cells characterized by specific expression values. For each GMM model, conditional probability is calculated and visualized in the form of lines. Therefore for each expression value, there is information about the likelihood that the value comes from a given component. Each component is transformed into a new feature that contains values between 0 and 1, indicating the belonging of each cell to that component. The features have a different distribution, where the values near one stand for cells that express the particular range of marker values. (Source: personal collection).

4.2.5 Membership function

Some artifacts must be corrected to make the results reliable and use conditional probabilities as new features. As it was mentioned, conditional probability is the likelihood of a marker's expression value occurring given the distribution with specified parameters. In other words, for each GMM component described by a set of parameters, the x expression value has a different probability of belonging to the component, and the sum of the components is equal to 1. It can be visualized as conditional probability lines indicating the probabilities for each component and the whole marker expression range (Figure 4.5.). Moreover, the first and last of the conditional probability lines should start and end at a level equal to one, respectively, since the points at both ends of the marker distribution have the smallest distance to the first and last of the components (probability equal to 1).

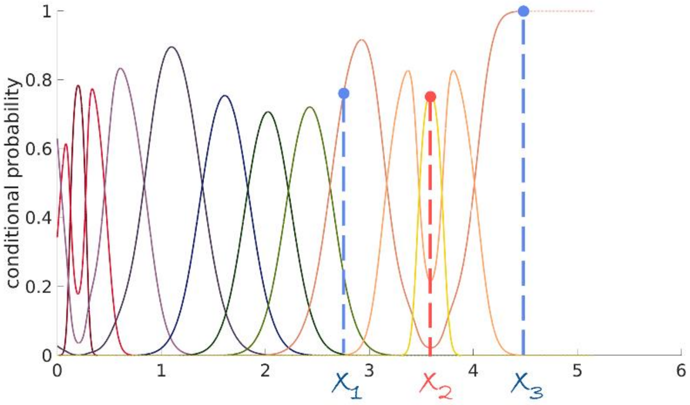


Figure 4.5. The main problem with conditional probabilities used to assign cells to GMM components.
Suppose one or more conditional probability lines dominate in more than one range of x-axis values. In that case, it may happen that two cells with very similar values of the marker will belong to different GMM components. At the same time, another cell with a higher difference in the expression value will belong to the same component. In the presented example, a cell with marker value x_1 belongs to the same GMM component as the cell with expression x_3 of the marker, but closer to the cell with marker value x_2 belongs to a completely different component. (Source: personal collection).

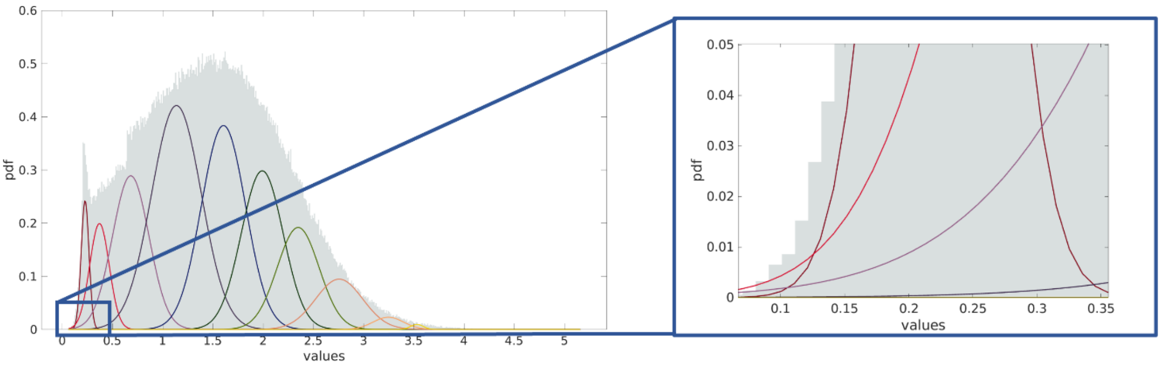


Figure 4.6. The source of the dominance of a component in different ranges of values. Example on CD45 marker from Samusik's dataset.
The image shows the zoomed area at the beginning of the marker's expression values distribution. As can be observed, the first peak (dark red, left panel), the first component with the smallest mean value, is dominated by light red and violet lines (right panel). As a result, the expression values from a range of approximately [0, 0.125] will have a higher probability of belonging to the light red or violet components that describe Gaussian distributions with higher mean values than dark red, which is the closest. The same problem may occur at the end of the marker distribution. (Source: personal collection).

Because the conditional probability lines are calculated from GMM components, some of them may have more than one peak (may dominate in more than one range of expression values). This is a known artifact of the Gaussian probability density function used in mixture models. When two Gaussians partially overlap, which is determined by the mean and standard deviation values, the one with a higher standard deviation will have a broader shape and eventually higher values on its ends behind the other Gaussian (Figure 4.6). The effect is visible on conditional probability lines in Figure 4.5. for example, the darker orange line has two peaks: around the x-axis values of 3 and 4.5.

Several steps have to be applied to create the proper expanded feature space:

1) The conditional probability lines must be corrected to have only one maximum (one peak) in the entire range of marker values.
2) The first and the last components' lines should be identified, and their dominance at the beginning and end of the distribution should be restored.
3) The first and the last conditional probability lines should start and end (respectively) at a probability value equal to 1, therefore, have a different shape modeled than the rest of the lines.
4) Some components may have too low component proportion (weight) and not dominate at any range of marker values. Therefore, these components bring nothing to the expanded feature space and should be identified and removed (Figure 4.7).
5) Correcting the conditional probability lines should not change the points of intersection of the verified lines. Intersections are often used as thresholds in various problems. Hence, the lines above the intersection points should be modeled carefully to reflect their original shape best.

The proposed correction algorithm written in MATLAB2022b meets all the required points outlined above, and it works directly on the conditional probability lines and their shapes. However, because the lines are modified after the correction, and the probabilities for each x-axis value do not sum up to 1, it cannot be further called "conditional probability." Therefore we proposed the name "membership function" since the values indicate the membership of observation to the components. The main parts of the pipeline are presented in Figure 4.8.

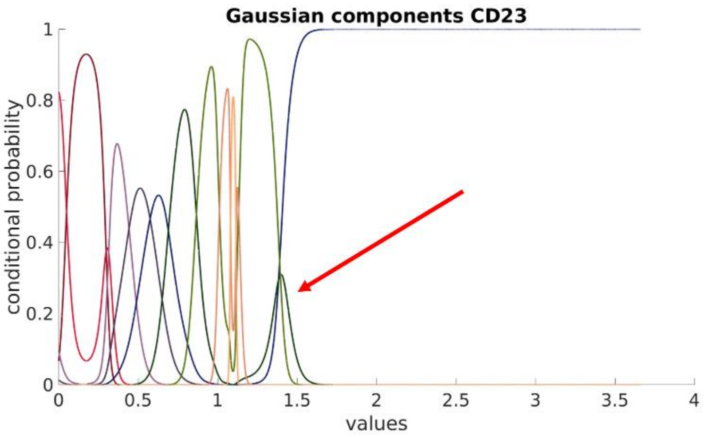
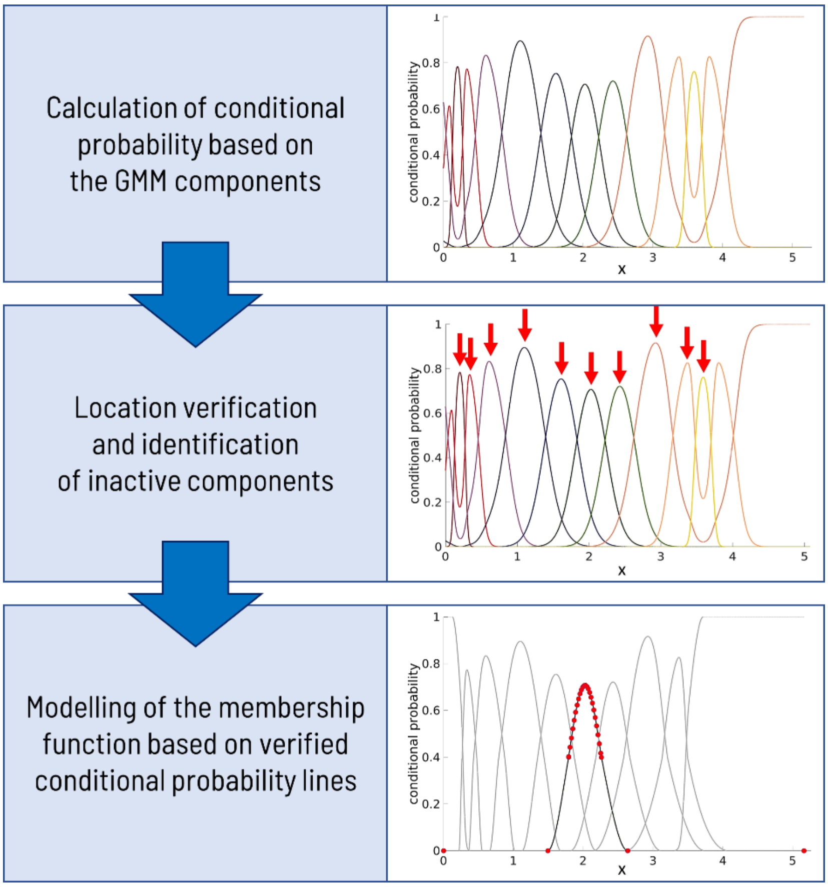


Figure 4.7. The inactive component in the model is visible in the conditional probability lines plot.
The red arrow points at the inactive component in the Gaussian Mixture Model. The component has too small a weight to dominate in any range of the marker values. Since the component brings nothing to the expanded feature space, it should be removed from the set of components. (Source: personal collection).



Calculation of conditional probability based on the GMM components

Location verification and identification of inactive components

Modelling of the membership function based on verified conditional probability lines

Figure 4.8. The pipeline of conditional probability lines correction.
Firstly, a GMM decomposition is performed for a given marker to get the mixture components. Based on the components, a conditional probability is calculated that can be visualized as a set of lines. Each line represents the probability for a given marker value that determines the assignment of the value to the component. Each peak shows a region in which the component dominates, which means the range of marker values that belong to that component. (Source: personal collection).

Firstly, the algorithm verifies whether a component's line has a peak near its mean value parameter, which is recognized as the real one. Secondly, it checks whether each of the components has a range of values in which it dominates the rest of the lines. Otherwise, the component is inactive and is removed from the model, and the probabilities are recomputed, to sum up to 1. In the next step, the line for the first component is corrected – the algorithm changes the shape of the line to start at point (0, 1), which is the maximum membership value for the marker expression of 0. It samples a set of points from the right side of the peak to model the line behavior for the higher expression values end eventually ends the line with a membership value of 0 for the maximum expression value. After the first component is improved, the middle components are modeled similarly, except that the lines' beginning and end have a membership level equal to 0. Finally, the last component is corrected to end at a membership level of 1 for the maximum marker value. The modeling of the lines is accomplished with *pchip* function and a set of points within the interval of m ± 3s (where m – mean value and s – standard deviation) as well as the maximum and minimum values of the whole expression range.

The resulting membership function assigns each x-axis value (marker expression) to one, the closest component (the highest membership value), and to the neighbor components with smaller membership values. The values' distribution is drastically different from the markers' original distributions (Figure 4.4.). Therefore, the existing identification approaches are expected to generate worse results. To be able to use the expanded feature space effectively, the algorithms have to be adapted to the characteristics of the domain. Thus, an original approach for cell-type identification is proposed in the doctoral thesis.

4.2.6 Binary transformation of the expanded feature space

Because the expanded feature space is continuous and may include minimal values that can influence the algorithms negatively regarding computation time, the binary representation of expanded feature space was proposed to speed up the process.

The binary representation of the expanded domain is based on the fact that each Gaussian Mixture Model component is described by mean and standard deviation values that translate to the conditional probabilities. Cells that had expression values falling within the range of mean ± 3 std for a given component of a given marker received a value of 1 for that component and 0 otherwise. Thus the result is a sparse matrix of 0 and 1 values indicating in which components the cell expression for a given marker is included.

4.2.7 Feature selection for the expanded domain

Since it is unlikely for each new feature in the expanded domain to contribute to each cell type profile that will distinguish them from others, the most important features should be selected before clustering. Those features are the most distinguishing features between groups of cells. But, because of the specific distribution of features in the expanded domain, most known and basic feature selection methods are not applicable in this case.

To find out which characteristics of the features may be valuable for feature selection, some of them were calculated using the original expanded domain and presented in the form of a histogram: mean, variance, the Relative Diversity Index (RDI) (4.3), and a ratio of the number of non-zero values to all values (NZR) (4.4). The best feature selection method was chosen empirically after applying a proper threshold.

Thresholds for the computed statistics were determined with the Gaussian Mixture Model decomposition of the distribution of each statistic into an optimal set of components found with the Bayes Factor. The cut-off value was found as the intersection point between the first and the second components. As a result of feature selection, features with a statistic value greater than the threshold are preserved in the analysis.

$$\mathrm{H} = \frac{\mathrm{Nlog(N)} - \sum_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{n_i log\,(n_i)}}{\mathrm{N}} \tag{4.1}$$

$$\mathrm{H_{max}} = \sqrt{\mathrm{k}} \tag{4.2}$$

$$\mathrm{RDI} = \frac{\mathrm{H}}{\mathrm{H_{max}}} \tag{4.3}$$

$$\mathrm{NZR} = \frac{|\mathrm{x} > \mathrm{min\,(x)}|}{\mathrm{N}} \tag{4.4}$$

Where: N – the number of cells; n_i – the number of cells in a category; k – the number of categories; x – feature values; |x>min(x)| - the number of values that are higher than zero (the minimum value of the feature).

The relative diversity index for binary transformation was calculated with two categories, 0 and 1. On the other hand, the continuous values had to be discretized into five categories defined by the uniform ranges of feature values.

After determining the best feature selection criterion, the same statistic was calculated for the binary expanded domain since the distribution is different from the original expanded domain. Therefore, applying the same threshold would not create accurate results.

4.2.8 Divisive approach

The feature space expansion and optimization through the feature selection may not be sufficient to reveal rare cell subpopulations. Good evaluation metrics are insufficient to conclude that the given model is the best if the number of found cell populations is lower than expected (assuming high certainty that more cell types are present). A second division of the clusters is proposed to prevent a situation when the number of identified cells is underestimated.

The second division is based on the cluster labels obtained from the first division that was performed with the PARC algorithm, using the expanded domain and feature selection. In the next step, for each cluster, a GAP statistic [51] determines if the group should be further divided. Two approaches are considered: the second division performed by PARC and the k-Means algorithm. If the cluster should be further divided, the division is performed with k-Means into the number of subclusters determined by GAP statistic or with PARC using its built-in algorithm for finding the optimal number of clusters. The second division is made in locally optimized space - before each division, a feature selection is conducted with the threshold found through GMM decomposition. Therefore each further division is optimized appropriately to the cluster content.

The PARC algorithm was run with the default parameters (the same as applied for the first division), and the k-Means distance measure was set to 'euclidean' for the original expanded domain, which is continuous, and 'hamming' for the binary expanded domain, which is dedicated to this type of data.

Because the k-Means algorithm is not able to process millions of cells that may constitute a cluster, the sampling had to be applied during the second division. Ten thousand randomly chosen cells were selected from each group and fed to the clustering algorithm. The rest of the cells were assigned to the closest of the determined clusters based on a distance metric (Euclidean or Hamming).

4.2.9 Analysis of the dataset from Stellenbosch University

All the proposed algorithms and transformations were applied to the dataset provided by partners from Stellenbosch University in RPA.

The dataset was preprocessed the same way as Samusik's dataset – the expression values were ArcSinh transformed using a co-factor of 5. The batch effect was corrected using the cyCombine method. The dataset was analyzed using the divisive approach; therefore, it was prepared in two variants: original expanded and binary expanded domains. UMAP embedding was generated for each dataset transformation, and the main cell populations were found using the PARC algorithm with the default parameters as the first division. Then the resulting clusters were further divided using PARC and k-Means algorithms. Finally, the best method was indicated and applied to the statistical analysis.

Since the annotations for the dataset are unknown, it is impossible to calculate all of the evaluation metrics listed in *Section 4.2.10*. Therefore only the unsupervised scores were used. The proportion of patient groups was calculated for each cluster and visualized in boxplots. To verify the normality of the groups' proportions in each cluster, a Kolmogorov-Smirnov test was used. A two-way ANOVA test was applied to demonstrate the differences between group proportions in each cluster. Together with p-values, an effect size was computed.

4.2.10 Evaluation of clustering results

With the described idea of cell subpopulations and the present heterogeneity of data, the clustering algorithm should not be "punished" for the overestimation of clusters. Therefore, a novel approach was proposed to find concordant cell subpopulations among clusters for intra-cluster heterogeneity credit and include it in evaluating clustering results. First, each cluster is identified as one of the defined cell subpopulations obtained after 2D GMM decomposition (See *Section 4.2.3.*). Then, the evaluation metrics are calculated, including the knowledge of the subpopulations.

For the model evaluation, various metrics were used. Six measures are supervised (Adjusted Rand Index, Adjusted Mutual Information, Homogeneity, Completeness, V-measure, and Fowlkes-Mallows Index), and two are unsupervised (Calinski-Harabasz Index and Davies-Bouldin Index). The details about them are provided below.

The unsupervised metrics are used in the doctoral thesis as the most critical model selection indicators. In the real-life problem of cell-type identification, expert annotations are usually not known and are impossible to obtain. In that situation, the user may only evaluate a model and score the clustering result with unsupervised metrics. Therefore the goal was to maximize the Calinski-Harabasz Index and minimize the Davies-Bouldin Index at each analysis step.

The rest of the metrics (supervised) are used to present the influence of including cell subpopulations in the clustering evaluation. Since the expert annotations are biased, and many authors suggested that the analysis should not rely on them too much, including supervised metrics in the results would be misleading. The supervised metrics evaluate how well the clusters match the manual labels, while the unsupervised ones measure how well the clusters are separated and defined. Also, the unsupervised metrics are calculated based on the original values of the dataset (regardless of the dataset transformation on which the predicted labels were created) to measure the real impact of the result on similar cell separation.

Calinski-Harabasz Index (CHI)

Calinski-Harabasz Index is also known as the Variance Ratio Criterion. The CHI is a ratio of the sum of between-cluster and within-cluster dispersions. The advantage of this metric is that it is unsupervised and fast to compute. The score is higher for dense, well-separated, and convex clusters [52].

- Range of values: [0, +∞],

- Interpretation: the higher the score, the better-defined clusters.

Davies-Bouldin Index (DBI)

DBI defines the average similarity of each cluster with the actual cluster that is the most similar to it. The similarity is a ratio of within-cluster to between-cluster distances. The measure is unsupervised with a minimum score of 0. The score tends to be higher for convex clusters [53].

- Range of values: [0, +∞),
- Interpretation: the lower the score, the better separation of clusters.

Adjusted Rand Index (ARI)

This is a measure of the similarity of two sets of clusterings: the ground truth and prediction that is based on the Rand Index (RI) but adjusted for the chance grouping of elements. The simplified version of calculating RI (4.1) and ARI (4.2) is presented below. It considers all pairs of samples between the clusterings and count the number of pairs that belong to the same or different cluster [54], [55]. The advantage of ARI is the interpretability of the result and no assumption on the cluster structure, so it is suitable for different clustering algorithms. On the other hand, as a supervised metric, it requires the knowledge of ground true clusters.

- Range of values: [-1, 1],
- Interpretation: -1 – especially discordant clusterings, 0 – random assignment; 1 – identical clusters.

$$\mathrm{RI} = \frac{\mathrm{TP+TN}}{\mathrm{TP+TN+FP+FN}} \tag{4.1}$$

$$\mathrm{ARI} = \frac{\mathrm{RI-E[RI]}}{\mathrm{max(RI)-E[RI]}} \tag{4.2}$$

Where:

TP – true positives; TN – true negatives; FP – false positives; FN – false negatives.

Adjusted Mutual Information (AMI)

Mutual Information (MI) score measures the agreement of two assignments (4.3), and the adjusted variant accounts for the chance. AMI solves the problem with MI which returns the higher value for a bigger number of clusters in the two clusterings regardless of whether the division is better. It requires the knowledge of the true labels but is independent of the absolute values of the labels [56].

- Range of values: [0, 1],
- Interpretation: 0 – random assignment; 1 – identical clusters.

$$\mathrm{MI} = \sum_{i=1}^{|\mathrm{True}|} \sum_{j=1}^{|\mathrm{Pred}|} \frac{|\mathrm{True_i} \cap \mathrm{Pred_j}|}{\mathrm{N}} \log \frac{\mathrm{N}|\mathrm{True_i} \cap \mathrm{Pred_j}|}{|\mathrm{True_i}||\mathrm{Pred_j}|} \tag{4.3}$$

$$\mathrm{AMI} = \frac{\mathrm{MI} - \mathrm{E[MI]}}{\mathrm{mean([H(True), H(Pred)]) - E[RI]}} \tag{4.4}$$

Where:

N – number of samples; |True| - number of true clusters; |Pred| - number of predicted clusters; $|\mathrm{True}_i|$ - number of samples in true cluster i; $|\mathrm{Pred}_j|$ - number of samples in predicted cluster j; $|\mathrm{True}_i \cap \mathrm{Pred}_j|$ - number of samples in true cluster i and predicted cluster j; H(True) – entropy for true clustering; H(Pred) – entropy for predicted clustering.

Homogeneity metric (Homog)

The homogeneity metric measures if a given cluster contains only samples of a single true class [57].

- Range of values: [0, 1],
- Interpretation: 0 – non-homogeneous assignment; 1 – perfectly homogeneous assignment.

Completeness (Compl)

The completeness measures if all samples from a given class are assigned to the same cluster [57].

- Range: [0, 1],

- Interpretation: 0 – non-complete assignment; 1 – perfectly complete assignment.

V-measure (V-score)

V-measure is the harmonic mean of homogeneity and completeness. The advantage of the measure is no assumption on the cluster structure. However, homogeneity, completeness, and v-measure are not normalized concerning the random assignment, therefore, may yield higher values when the number of clusters is large [57].

- Range of values: [0, 1],
- Interpretation: 0 – bad assignment; 1 – a perfect score.

$$\mathrm{V} = \frac{(1+\beta) * \text{homogeneity} * \text{completeness}}{(\beta * \text{homogeneity} + \text{completeness})} \tag{4.5}$$

Where:

β – defaults to 1.

Fowlkes-Mallows Index (FMI)

This is a geometric mean of precision and recall. In the case of a completely random assignment, the FMI will be 0. No assumptions are made about the cluster structure. Requires the knowledge of true labels [58].

- Range of values: [0, 1],
- Interpretation: 0 – bad, random assignment; 1 – significant agreement of clusters.

$$\mathrm{FMI} = \frac{\mathrm{TP}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})}} \tag{4.6}$$

Where:

TP – true positives; FP – false positives; FN – false negatives.

4.3 Results for Samusik's dataset

This section presents the results of the analysis of Samusik's dataset. The dataset was preprocessed, the expression values were ArcSinh transformed with a co-factor of 5, and the batch effect was corrected using the cyCombine method. In the next step, cell-type subpopulations were determined to include the data heterogeneity during evaluation. The most efficient and popular clustering methods, according to the literature, were applied for cell population identification. A new approach was proposed that is based on creating expanded feature space (called the expanded domain) with the different transformations of the values and feature selection method. The modified data were applied to PARC clustering for comparison. Finally, a new divisive approach was proposed that was tuned on Samusik's dataset in the chosen transformations of expanded feature space. The following subsections describe the steps in detail.

4.3.1 Data heterogeneity – cell subpopulations

The decomposition of samples from each cluster of the Samusik dataset in UMAP space with a two-dimensional Gaussian Mixture Model resulted in a set of potential subpopulations that were further filtrated.

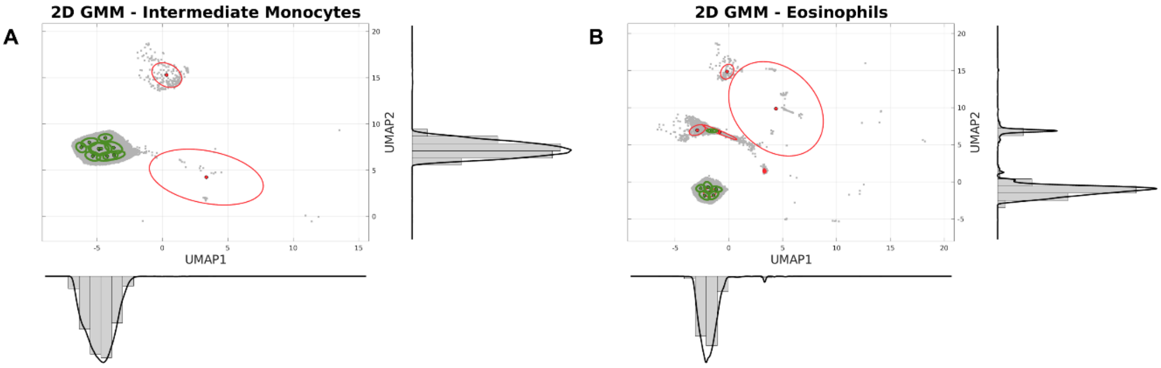


Figure 4.9. Verification of subpopulations that resulted from Samusik's cell types after 2D GMM.
Examples showing 2D GMM decomposition of Intermediate Monocytes and Eosinophils. Green components indicate subpopulations that were considered authentic and red components were discarded from the analysis because of too small component proportion. Side histograms show the distribution of the first and second UMAP values and identify distant subpopulations with separate peaks. (Source: personal collection).

Figure 4.9. shows exemplary results of the decomposition of two different cell types. The red ellipses represent components that were filtered out since their component proportion (weight) was too small. The threshold was found through 1D GMM decomposition of weights belonging to all cluster components and was equal to 0.064. Therefore all components containing less than 6,4% of the dataset were excluded, and the observations were assigned to the next closest component in the feature space. The green ellipses represent the subpopulations that remained in the analysis. Interestingly, most of the subpopulations are created in the primary cluster location, but there are cases when a quite representative subpopulation appears further in the UMAP space, like in the example of Eosinophils (Figure 4.9.B). The side histograms also confirm this.

The verified final subpopulations are presented in Figure 4.10. with a color indicating the parent cell population. The total number of subpopulations was 90. The way each cluster is split into subpopulations is the assumed acceptable division. Thus the results from clustering algorithms are compared to the subpopulations and identified as belonging to one of them. This is a novel approach that tries to take into account the possibility of human errors during the manual annotation of the data. During the model evaluation, the subpopulation influences the calculated metrics. Therefore the model is not "punished" for overestimating the number of clusters since the found clusters contain distinct cell subpopulations.

The clusters are assigned to the closest subpopulation based on the smallest Euclidean distance. The distance is measured between the clusters' centroids and the centroids of defined subpopulations in the marker space. Therefore the subpopulation assignment is independent of the UMAP projection.

4.3.2 Comparison of existing solutions in the regular domain

The preprocessed Samusik dataset was used to compare the existing methods for cell-type identification. However, because of the large number of cells (514,386), it was not possible to run all of the methods mentioned in the introduction. Therefore the three best methods, according to the literature, were used: FlowSOM, ClusterX, and PARC. Unfortunately, PhenoGraph, also considered one of the best clustering methods, failed to process the dataset.

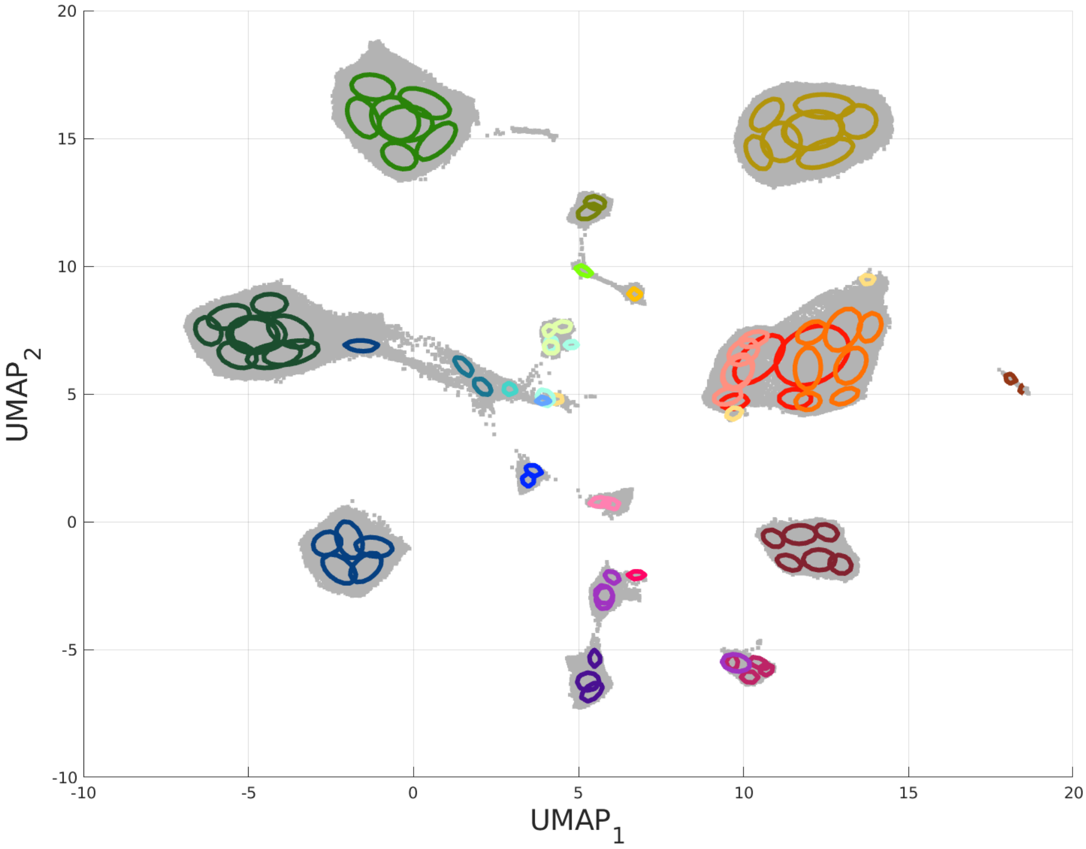


Figure 4.10. Found subpopulations of the known cell types in data from Samusik *et al.*
Each subpopulation originating from the know cell types is represented as an ellipse with a shape determined by the mean values (centroid) in the UMAP space and a covariance matrix. The color indicates the parent cell type. (Source: personal collection).

For the model evaluation, the chosen metrics were calculated twice: in a traditional manner and considering the found subpopulations of each cell type. The resulting cluster assignment is presented in Figure 4.11. using mISO plots. Details and calculated metrics are summarized in Table 4.2.

It can be observed (Figure 4.11.) that each clustering algorithm divides the cells differently, although they identified almost the same number of clusters. It is visible looking at the aggregate of cells containing the orange and red groups of cells or the one with the dark red group of cells. However, none of them identified the rare subpopulations in the middle of the UMAP plot. FlowSOM failed to identify a large group of cell population – Intermediate Monocytes, the method joined them with Classical Monocytes. The results show that the clustering algorithms are sensitive to the heterogeneity of data and consider the differences of each cell population to be significant enough to create two separate clusters from one that an expert identified. On the other hand, the marker profiles of the rare cell populations are not strong enough to be distinguished. Therefore further analysis tries to find a solution for the rare cell subpopulations focusing on extracting more detailed marker profiles.

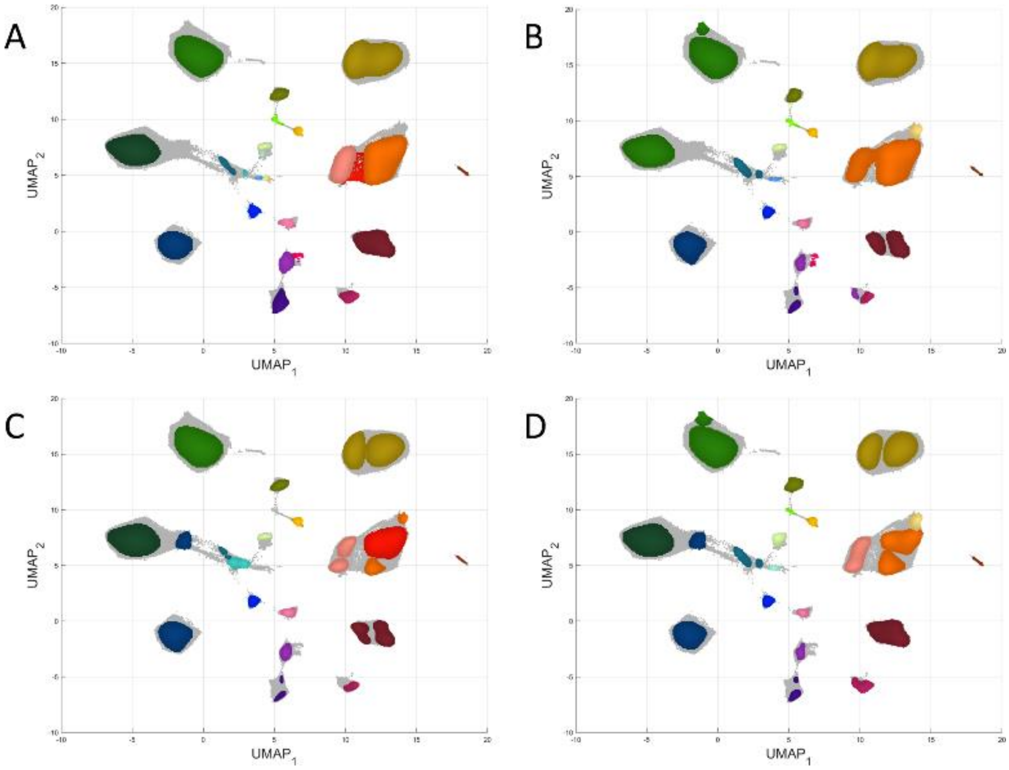


Figure 4.11. Results for the three existing cell-type identification approaches for a dataset in the regular domain.
Each cluster was compared to the found cell subpopulations to identify the cell types. The colors indicate the parent cell type. A) True assignments provided by experts. B) 24 clusters identified by FlowSOM. C) 25 clusters identified by CusterX. D) 24 clusters identified by PARC. (Source: personal collection).

Table 4.2. A detailed summary of the clustering results for a dataset in the regular domain.
The table presents calculated metrics for the chosen existing clustering methods and compares the scores before and after taking into account the cell subpopulation.

Standard approach (without knowledge about cell subpopulations)									
Algorithm	**#clusters**	**ARI**	**AMI**	**Homog**	**Compl**	**V-score**	**FMI**	**CHI**	**DBI**
FlowSOM	24	0.7030	0.8390	0.7869	0.8985	0.8390	0.7691	62202.22	**1.4728**
ClusterX	25	**0.8217**	0.8749	0.9292	0.8265	0.8749	**0.8504**	68436.24	1.6724
PARC	24	0.8189	**0.8944**	**0.9506**	**0.8447**	**0.8944**	0.8493	**77942.28**	1.5263
Including cell subpopulations									
Algorithm	**#clusters**	**ARI**	**AMI**	**Homog**	**Compl**	**V-score**	**FMI**	**CHI**	**DBI**
FlowSOM	24	0.7096	0.8473	0.7815	0.9252	0.8473	0.7767	73251.44	1.4552
ClusterX	25	0.9475	0.9197	0.9220	0.9176	0.9198	0.9547	78742.18	1.4573
PARC	24	**0.9646**	**0.9440**	**0.9438**	**0.9442**	**0.9440**	**0.9695**	**82305.73**	**1.4162**

After examination of the values from Table 4.2., it can be noticed that the score increased after including the knowledge of cell subpopulations. The best scores in both scenarios were obtained for the PARC clustering algorithm; therefore, only this method was used in further analysis. Moreover, further analysis scores are presented for the case with included cell subpopulations as the better measure for goodness of clustering, taking data heterogeneity into account.

4.3.3 Process of generating the expanded feature space

Each marker was applied to Gaussian Mixture Model decomposition to create the expanded feature space (dataset in the expanded domain). As a result, a set of optimal components was established for each marker using BIC. The number of resulting components varied from 2 to 13.

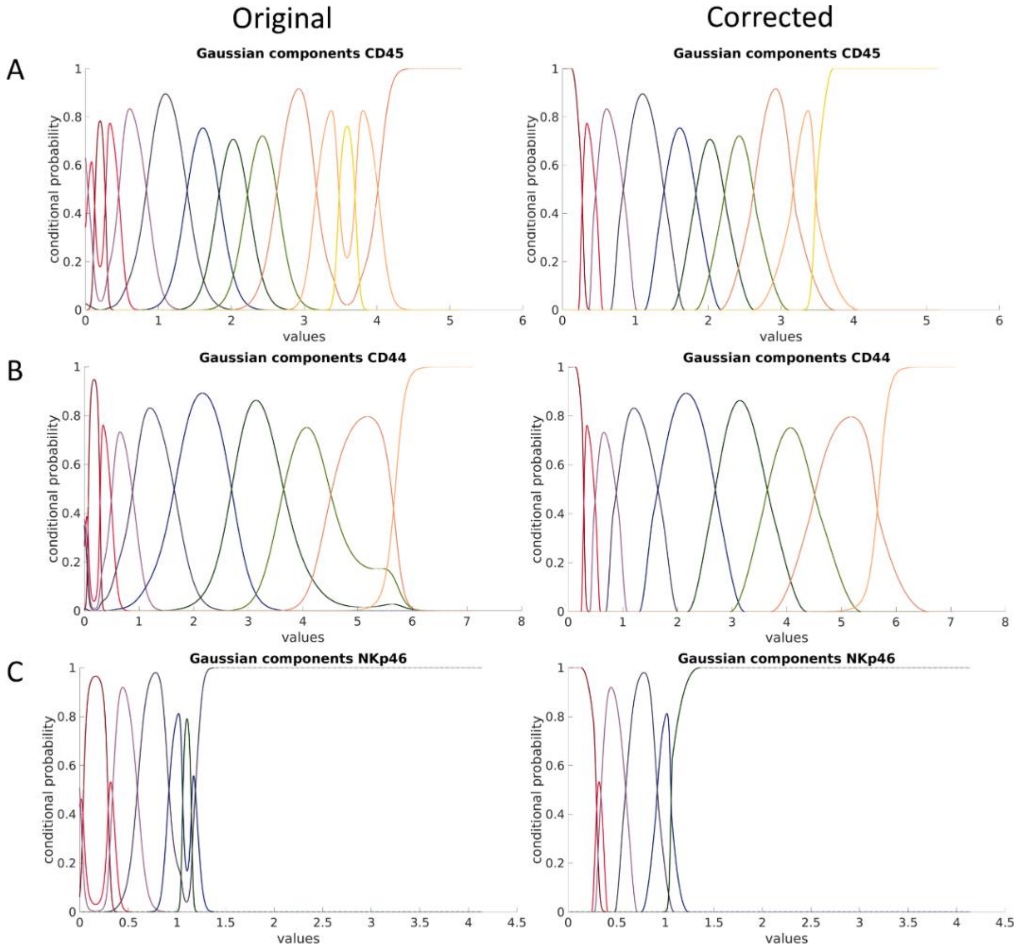


Figure 4.12. Exemplary conditional probability lines before and after correction (membership function). Effect of conditional probability lines for A) CD45 marker. B) CD44 marker. C) NKp46 marker. (Source: personal collection).

In the next step, conditional probability lines were computed and corrected. The conditional probability lines before and after correction are visualized in Figure 4.12. It can be noticed that the correction algorithm meets the assumptions – it verifies the placement of real peaks, finds the first and last component lines and corrects their shape to begin/end with a y-axis value of 1. It resembles the top part of the middle conditional probability peaks very well and preserves the intersection points of the verified lines. After the correction, the values no longer represented probabilities, so the transformation was called the membership function.

The new feature space consisted of 318 new features containing values from 0 to 1 that indicate the membership of a cell to a specific component in the marker's GMM model based on the original expression value. The expanded data domain can help extract valuable information about rare cell populations. This can be observed in Figure 4.13. that shows a comparison of the Sca1 marker for each known cell type where the component filled with color represents the assignment of the group of cells to the component based on the highest membership values. As presented, most cell types either express high values or low values for the marker and belong to the first and last of the component, except one cell type: HSC, which belongs to the central component. The cell type is one of the rare populations.

A different marker profile can be observed comparing other markers between the cell types, as presented in Figure 4.14. The figure contains chosen markers for five different cell types (the number of presented cell groups and markers is limited by space). Since each component is a separate feature in the expanded domain, different features contribute to the cell type marker profile.

4.3.4 Results for the dataset in the expanded domain

The prepared and corrected expanded feature spaces representing each marker component's membership values were applied to the existing solutions to see if the expansion of dimensionality helped the cell identification process. Moreover, UMAP transformation and PARC clustering were generated with original expanded domain values and binary representation of the space. Despite the new UMAP projection, the results for clustering methods are presented on the regular domain UMAP space for better comparison purposes.

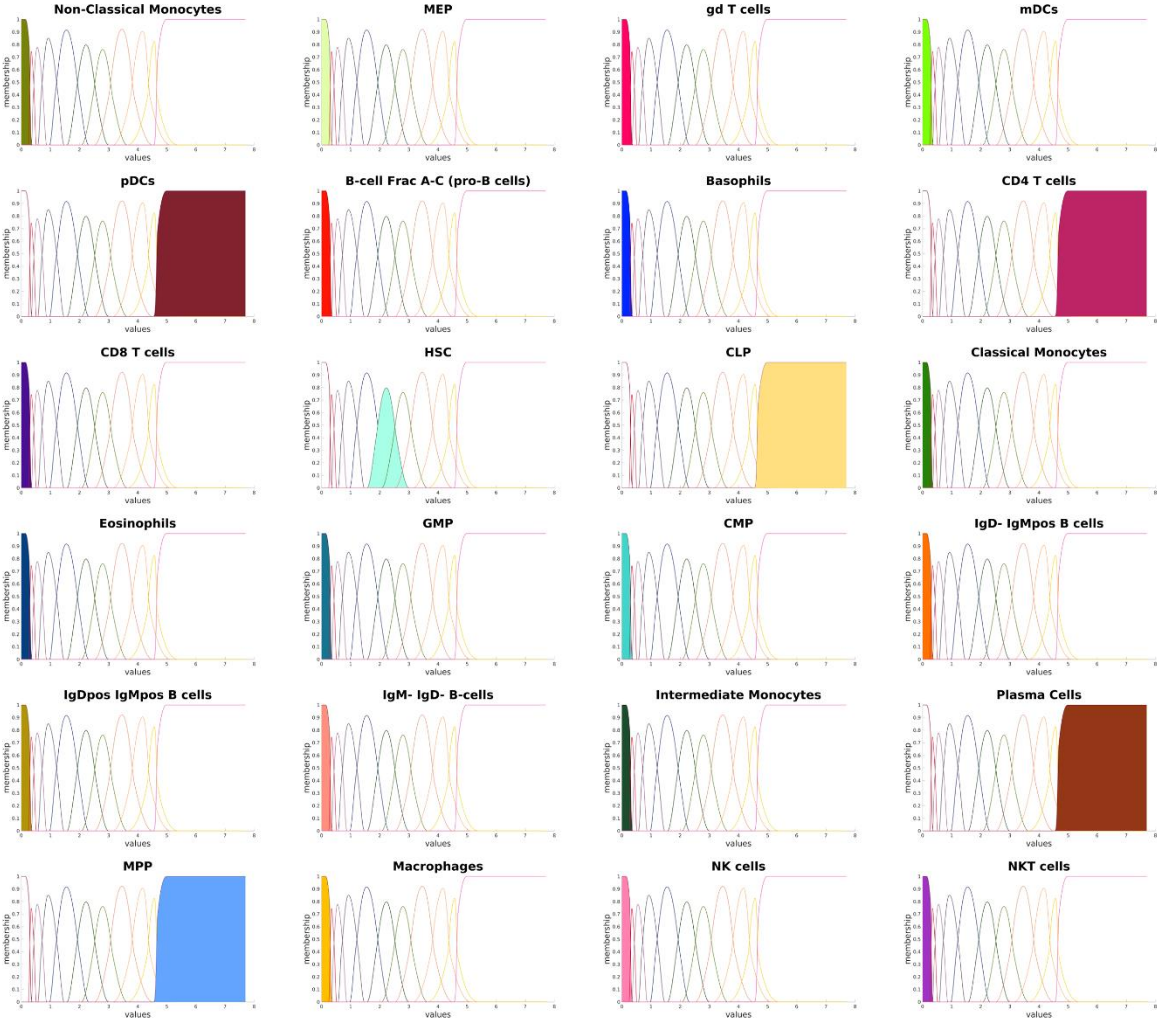


Figure 4.13. Visualization of the expanded domain idea on the Sca1 marker example.
Each plot represents the component of the highest membership (90% percentile) for each cell type in Samusik's dataset. Most cell types express values represented by the first or the last component (the highest and the lowest marker expression values). Still, one cell type (third row and second column), also a rare population, expresses values from the middle of the marker's range. In the expanded domain, the rare population has a distinct feature that may help in the identification process. (Source: personal collection).

4.3.5 Comparison of UMAP projections

Figure 4.15. represents the dataset in the UMAP space, thus the influence of the expanded domain on the data structure. In the visualization of UMAP embedding, the placement of cells and clusters differ significantly. However, there are also similarities, like two similar clusters being close to each other regardless of the transformation. Although the original expanded feature space preserves the global dataset structure well, the small subpopulations tend to overlap. As can also be observed, some of the clusters in expanded space and after binary transformation (Figure 4.15.B-C) have become more similar and lie closer than in the regular space (Figure 4.15.A).

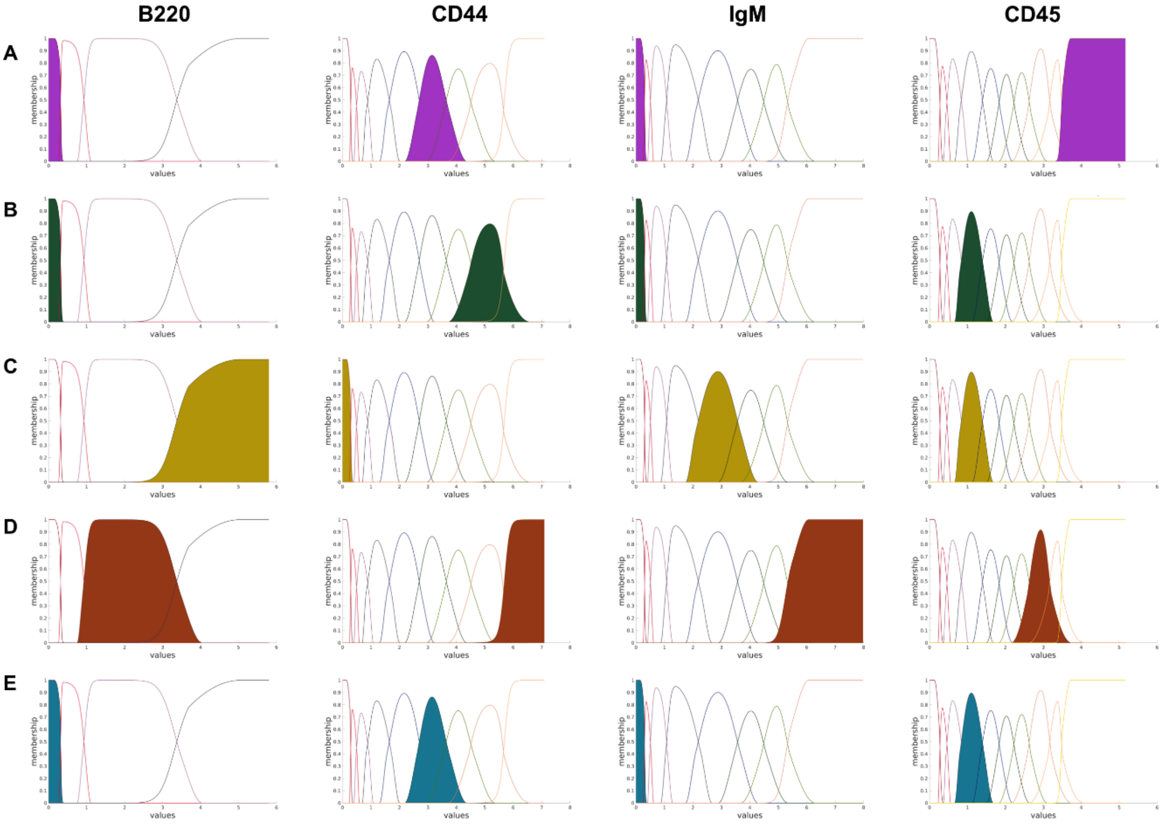


Figure 4.14. Comparison of five chosen cell types in the expanded domain.
Five cell types and four different markers. The highest membership value decides which of the marker's components the cell type express (determined by the maximum value of 90% percentiles for each component). Since each of the components creates a new feature in the expanded domain, they provide information about different cell populations. A) NKT cells. B) Intermediate Monocytes. C) IgD+ IgD+ B cells. D) Plasma Cells. E). GMP. (Source: personal collection).

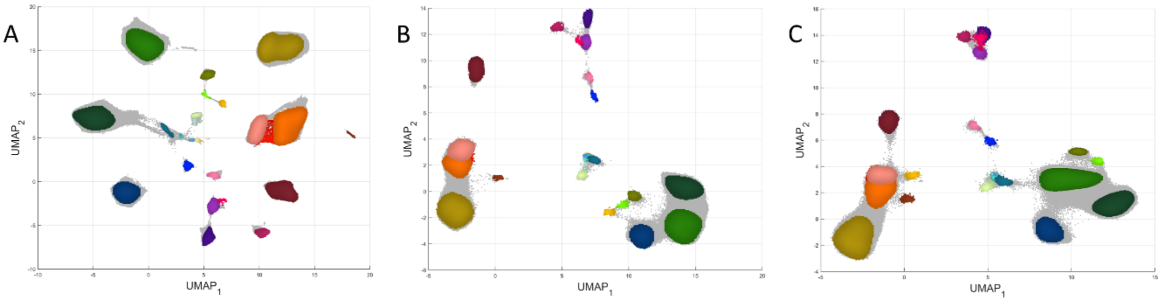


Figure 4.15. UMAP projection for different dataset transformations.
A) Regular domain of the dataset. B) Original expanded domain of the dataset. C) Binary representation of the expanded domain. (Source: personal collection).

Comparing the regular to the original expanded feature domain, it is visible that in the expanded space, the biologically similar cells are closer than in the regular one. The effect is clearly visible in Figure 4.16. Moreover, the groups of similar cells are more separated in the expanded feature space. Other cell types are placed near the other types that are the most similar in a biological context.

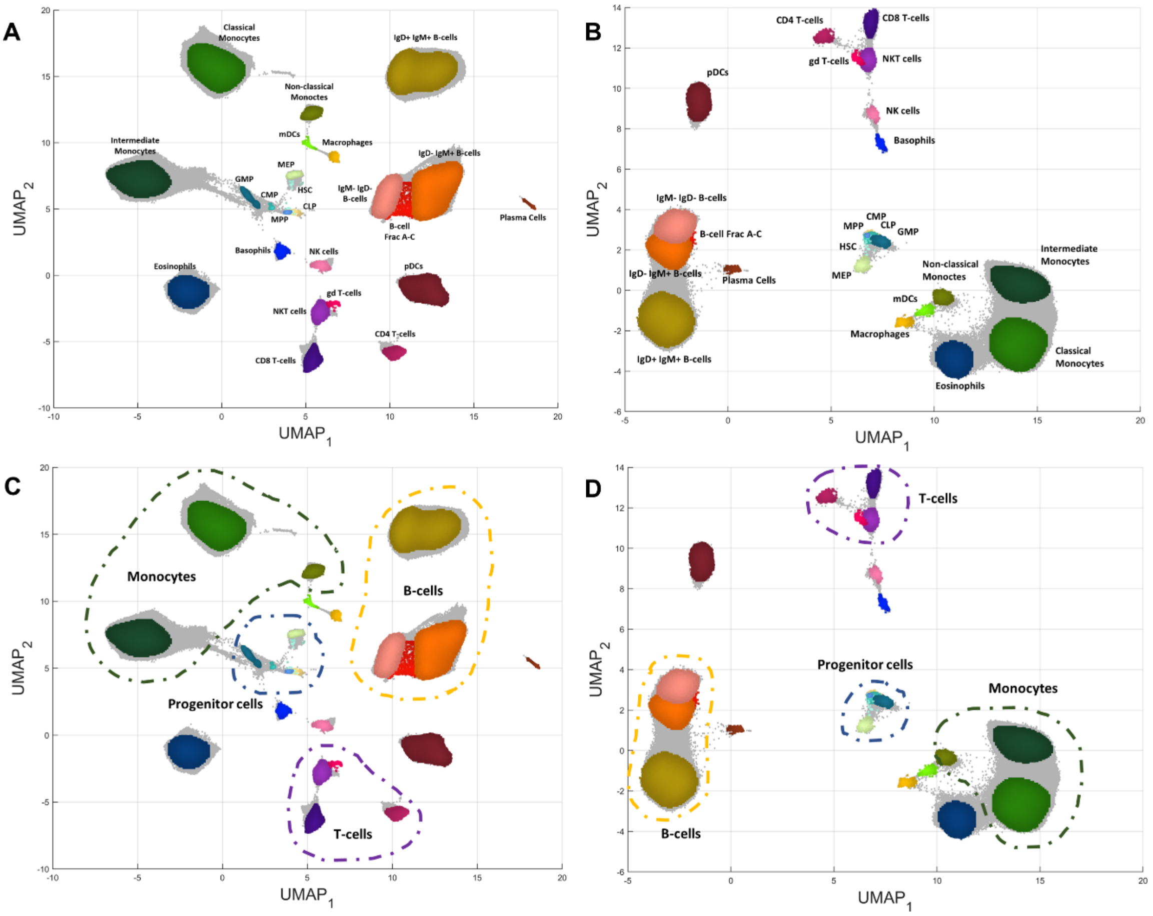


Figure 4.16. UMAP projection of regular and expanded feature space and the influence on cell type's position.
A) Regular feature space with the annotations provided by experts. B) Expanded feature space with the annotations provided by experts. The position of the cell groups is different than in the regular space, but the similarities between the types are preserved. C) Main groups of similar cell types in the regular domain. D) Main groups of similar cell types in the expanded domain. It is visible that the cell types within the groups are closer than in the regular domain. The rest of the cell types are placed near similar cells in the biological context. (Source: personal collection).

4.3.6 Comparison of existing solutions between expanded and regular domains

The proposed original (without further transformations) expanded feature space was applied to cell population identification with the selected existing methods: FlowSOM, ClusterX, and PARC. The results are presented in Figure 4.17. and summarized in Table 4.3, along with the results for the regular domain of the dataset.

Using the expanded feature domain resulted in a better definition and separation of the cell clusters based on the Calinski-Harabasz and Davies-Bouldin Indices. The results are also visually compared with the regular domain; however, the number of found cell populations is smaller.

Two aspects must be taken into account when analyzing the above results. First, the existing solutions are not adapted to the data distribution characteristic of the proposed expanded domain. Therefore they may not work optimally. Second, the clustering was conducted on the entire dataset, meaning the complete feature set. Still, not all of them may provide sufficient information to contribute to the separation decision. Therefore, although the results are better regarding data structure and homogeneity, feature selection methods may be beneficial.

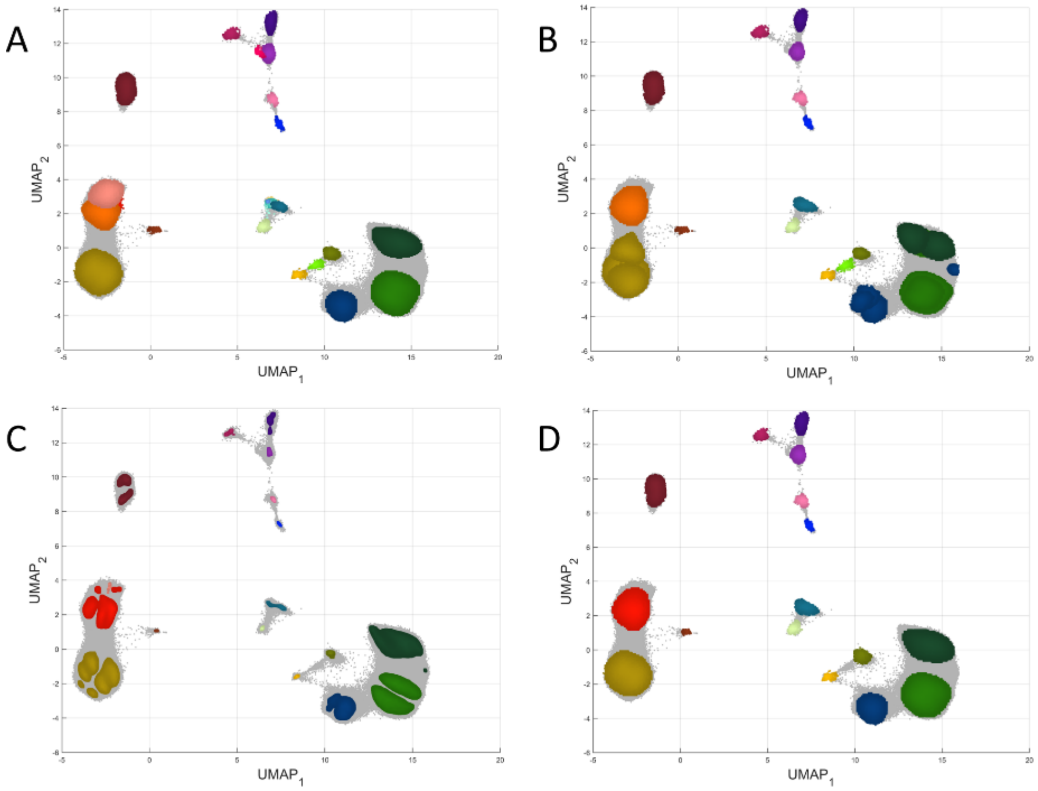


Figure 4.17. Results for the three existing cell-type identification approaches for a dataset in the expanded domain.
A) True assignments provided by experts. B) 24 clusters identified by FlowSOM. C) 19 clusters identified by CusterX. D) 20 clusters identified by PARC. (Source: personal collection).

Table 4.3. Comparison of the results from chosen existing methods in regular and expanded domains.

		Regular domain		Expanded domain	
Algorithm	**#clusters**	**CHI**	**DBI**	**CHI**	**DBI**
FlowSOM	24	73251.44	1.4552	**88863.82**	**1.3356**
ClusterX	19	78742.18	1.4573	**90746.60**	**1.3944**
PARC	20	82305.73	1.4162	**95594.57**	**1.3168**

4.3.7 Clustering results for expanded feature space

The created and corrected expanded feature space, and its binary representation were applied to PARC cell population identification. The model was run with the same parameters each time. Results were visualized using the UMAP space for the appropriate feature domain.

The clustering method found 20 and 14 populations for original expanded and binary transformations of the expanded feature space, respectively. The evaluation metrics were calculated considering the identified cell type subpopulations and are presented in Table 4.4. The best results were obtained for the binary transformation of the expanded feature space. However, based on the expert's annotations, the number of clusters was smaller than expected. The visualizations (Figure 4.18.) reveal that the small subpopulations remained unidentified, especially for the binary expanded feature space.

Table 4.4. A detailed summary of the clustering results for the dataset in the expanded domain.
The calculated evaluation metrics include information about cell subpopulations.

Transformation	#clusters	CHI	DBI
Original	20	95594.57	1.3168
Binary	14	**102633.50**	1.2984

Since the higher number of features in the expanded domain may influence the results, if not all of them contribute to the identification process, it was decided to apply the feature selection method before feeding the data into the model.

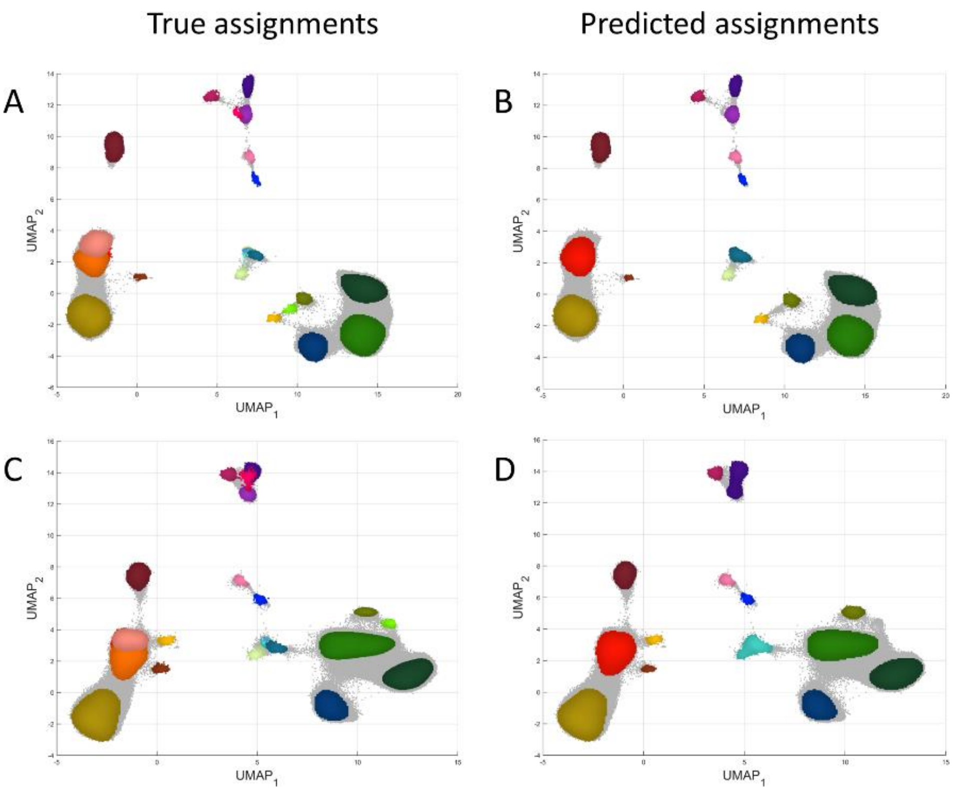


Figure 4.18. PARC clustering results for expanded feature domain in different data transformations.
The clustering results are presented in the UMAP projection of the appropriate feature space. A) True assignments in the original expanded space (membership values). B) Predicted clusters in the original expanded domain. C) True assignments in the binary expanded feature space. D) Predicted assignments in the binary expanded feature space. (Source: personal collection).

4.3.8 Clustering results for expanded feature space with feature selection

The feature filtration criterion was established using the original expanded feature domain and adapted to the binary transformation.

For each feature, four statistics were calculated: mean, variance, Relative Diversity Index (RDI), and a ratio of non-zero values to all values (Figure 4.19.). The characteristics were used for feature selection with an appropriate cut-off value as a criterion for discarding features. The best results were obtained using the relative diversity index (RDI).

A selection method was applied to the values in the expanded domain using the threshold for RDI measure equal to 0.17. The goal of feature selection was to filter out features that may negatively influence the results since not all of the markers' components may contribute to the expression profile of the cell populations.

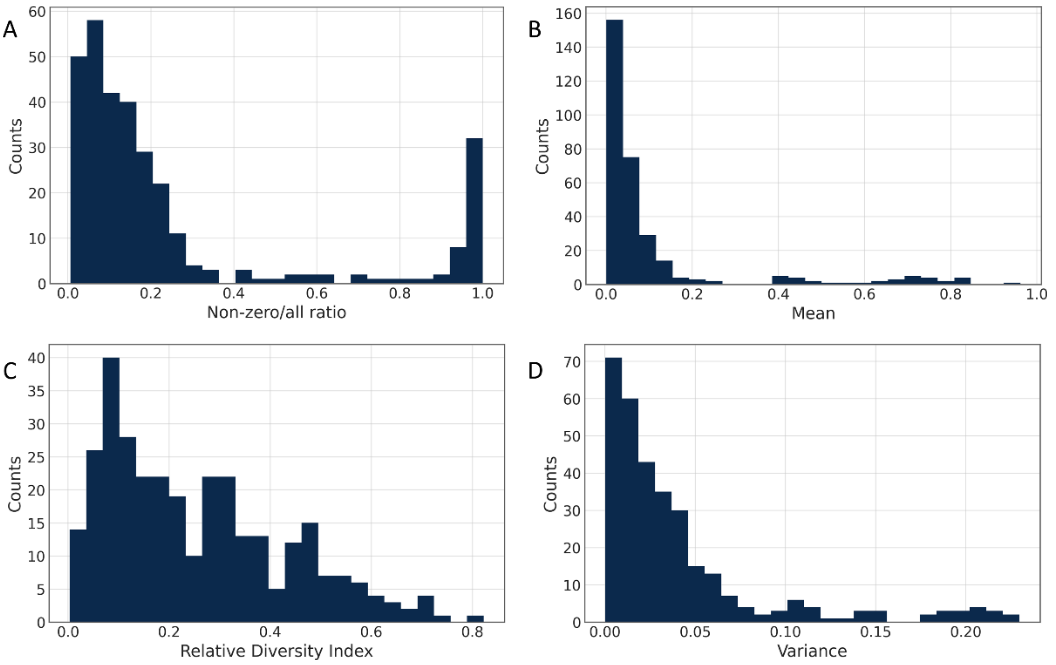


Figure 4.19. Characteristics of expanded feature domain.
The statistics were calculated for each feature in the expanded domain (without transformation). A) The number of non-zero to all values. B) Mean values. C) Relative diversity index values. D) Variance values. (Source: personal collection).

The Gaussian Mixture Decomposition established the threshold with Bayes Factor as the stopping criterion. The decomposition of RDI values resulted in two components, and their intersection indicated the cut-off value. For the binary expanded feature space, the threshold was established based on RDI values calculated for features in that space in a similar manner.

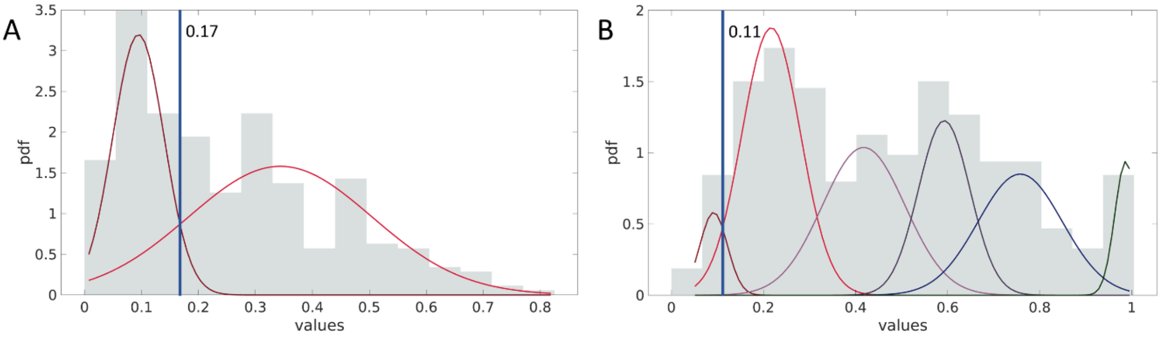


Figure 4.20. Decomposition of Relative Diversity Index values for the expanded domain features and the cut-off values.
A) Probability density functions for the two GMM components in the original expanded feature space. The blue vertical line indicates the intersection point between the lines and the cut-off value of 0.17. B) The decomposition of RDI values for the binary expanded domain with the vertical blue line representing the cut-off value of 0.11 as the intersection between the first two components. (Source: personal collection).

The PARC clustering algorithm was applied to each variant of the dataset with the same parameters as before. Figure 4.21. presents the results on UMAP embedding in the regular feature domain for better comparison with other results.

The clustering algorithm was able to find 13 and 11 cell populations in the original expanded and binary transformations of the expanded feature space after feature selection, respectively. The summary of the results is presented in The clustering results are presented in the UMAP projection of the appropriate feature space. A) True assignments in the original expanded space (membership values). B) Predicted clusters in the original expanded space. C) True assignments in the binary expanded feature space after feature selection. D) Predicted assignments in the binary expanded feature space after feature selection. (Source: personal collection).

Table 4.5.

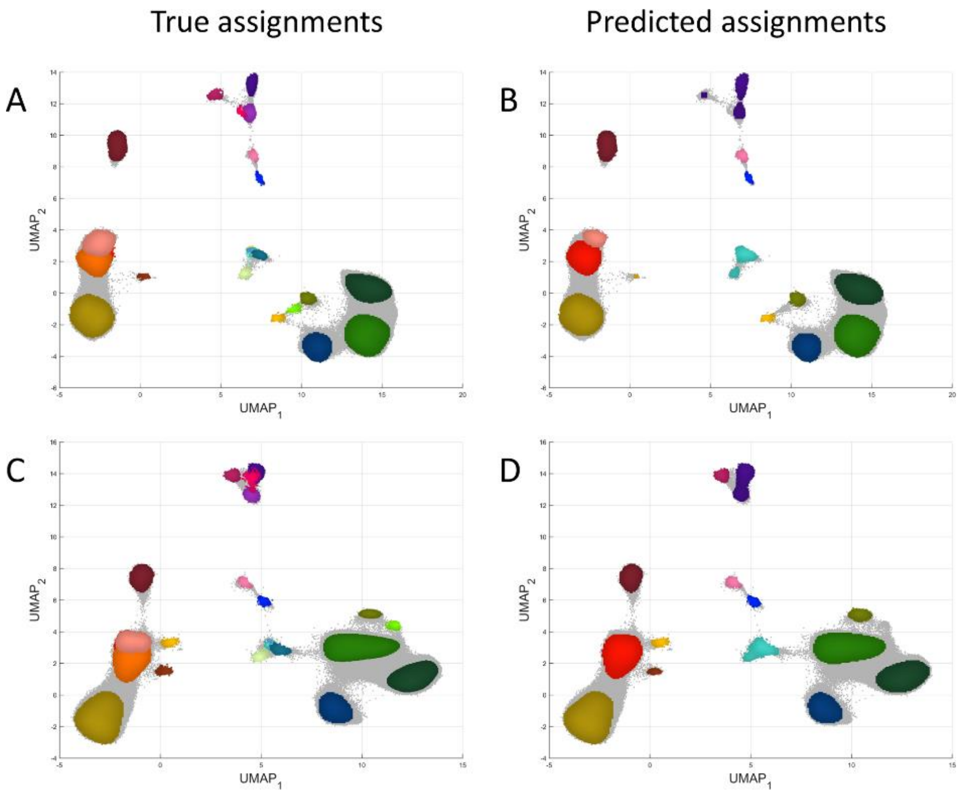


Figure 4.21. PARC clustering results for expanded feature domain after feature selection.
The clustering results are presented in the UMAP projection of the appropriate feature space. A) True assignments in the original expanded space (membership values). B) Predicted clusters in the original expanded space. C) True assignments in the binary expanded feature space after feature selection. D) Predicted assignments in the binary expanded feature space after feature selection. (Source: personal collection).

Table 4.5. A detailed summary of the clustering results for the dataset in the expanded domain after feature selection.
The calculated evaluation metrics include information about cell subpopulations.

Transformation	#clusters	CHI	DBI
Original	13	**105392.04**	1.5606
Binary	14	103723.85	**1.3079**

The best metrics were obtained for the original expanded feature space in terms of the Calinski-Harabasz Index and binary transformation in terms of the Davies-Bouldin Index. However, all the DBI values are slightly lower than the results without feature selection for the same dataset transformation. Feature selection caused the algorithm to find fewer clusters for the original expanded space, but the result almost did not change for the binary expanded domain.

Since the expanded feature space, applied transformation, and feature selection method were not sufficient to identify the cell types defined by experts or their subpopulations, it was decided to introduce another way for cell-type identification. This divisive approach will examine each found cluster separately and determine if it should be further divided.

4.3.9 Clustering results – divisive approach

The clustering results of the original and binary expanded domains (Table 4.5) are very similar. Therefore it was decided to check the results for both dataset transformations after the second division.

The divisive approach consisted of the PARC division and k-Means division or, again, the PARC division of each cluster. The first PARC division was conducted using the locally optimized expanded feature space, and the results were described in the previous section. The resulting clusters were applied to the second division determined by the GAP statistic. The GAP statistic also indicated how many groups should be created from the parent cluster using the k-Means algorithm. Before any split, each cluster was applied to feature selection with a threshold of 0.17 (original expanded domain) or 0.11 (binary expanded domain) (Figure 4.20.B). Therefore the feature space was optimized for each cluster before deciding if the group should be split and into how many subclusters. The PARC algorithm was run with default parameters and 'jac_weighted_edges' set to False for speed up (recommended by the authors). The k-Means method was run with distance metric set to 'euclidean' for continuous variables and 'hamming' for binary.

The algorithm decided to divide almost all the clusters further. The divisions produced 2 to 7 new subpopulations; therefore, the final number of groups was 27 to 41. Figure 4.22 shows the second divisions of the algorithm, where the divisions are presented in both the original and binary expanded domains. Table 4.6. presents a detailed summary of the obtained results.

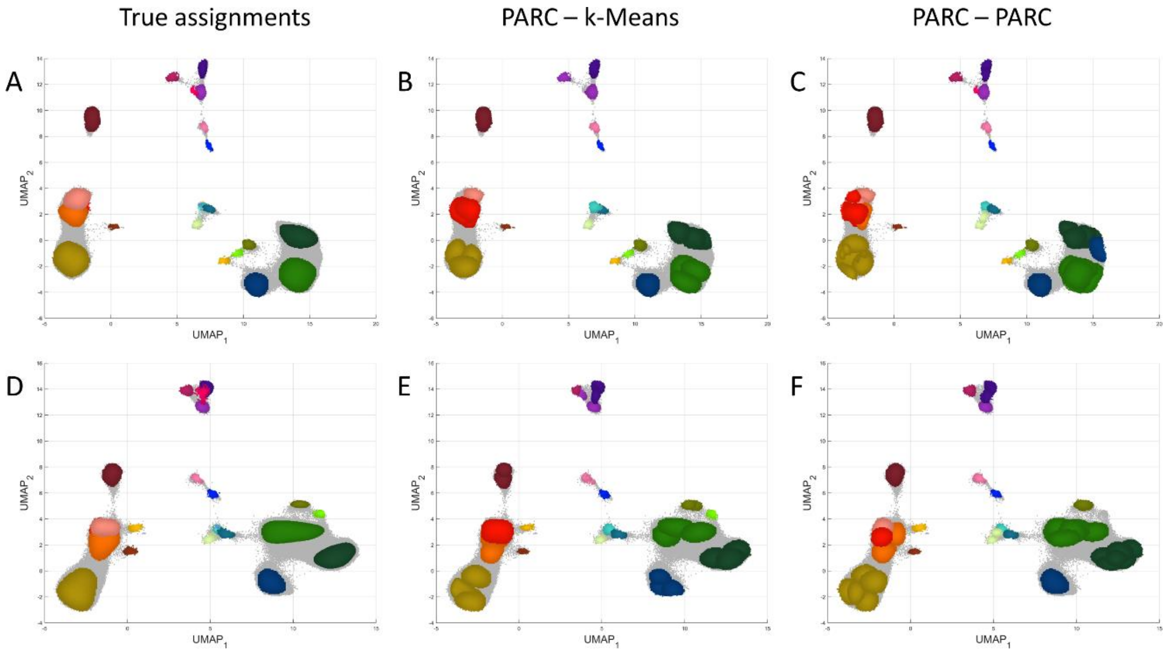


Figure 4.22. Final results for Samusik's dataset cell-type identification.
The first row presents results for the original expanded space, and the second row for the binary expanded feature space. A) True assignments provided by experts on UMAP projection of the original expanded domain. B) Locally optimized second division in the original expanded domain using the k-Means algorithm. C) Locally optimized second division in the original expanded domain using the PARC algorithm. D) True assignments provided by experts on UMAP projection of the binary expanded domain. E) Locally optimized second division in the binary expanded domain using the k-Means algorithm. F) Locally optimized second division in the binary expanded domain using the PARC algorithm. (Source: personal collection).

Table 4.6. Results for different two-step clustering approaches for Samusik's dataset.

Transformation	1st division	2nd division	#clusters	CHI	DBI
Original	PARC	k-Means	27	**94581.71**	1.4415
	PARC	PARC	34	75348.24	2.1048
Binary	PARC	k-Means	28	89442.34	**1.3957**
	PARC	PARC	41	82203.24	1.8460

Table 4.7. presents the evaluation scores for all the steps leading to the final result in the original expanded domain, which generated the best results. The Calinski-Harabasz Index measure was the major score in the evaluation of the algorithms; the second most important information was the number of clusters. Although the middle stages of analysis (PARC clustering with and without feature selection) have a higher CHI than the final stage, the number of identified cell populations is too low. After the second division, the metrics are still better than the results for the regular domain.

Table 4.7. Final results for Samusik's dataset and comparison with previous steps for the original expanded domain.

Metric	Regular domain	Full original expanded domain	Original expanded domain with feature selection	Second division with original expanded domain (k-Means)
CHI	82305.73	95594.57	105392.04	94581.71
DBI	1.4162	1.3168	1.5606	1.4415
#clusters	24	20	13	27

The generated visualizations of the clustering results (Figure 4.22.) reveal that the final algorithm found additional subpopulations from the main cell types defined by experts (for example, the yellow and green clusters). Still, some cell populations were not discovered, like those overlapping in the original expanded feature space (in UMAP projection, Figure 4.22.D).

Using PARC as a second step resulted in many identified cell populations that tend to overlap in the UMAP space and, overall, based on CHI and DBI scores, are worse than the results obtained with k-Means. On the other hand, using PARC allowed for identifying gd T-cells (dark pink color at the top part of Figure 4.22.C). The overlapping, however, is not a big problem since it may result from the visualization of the labels in the UMAP projection that was generated using all the features. In contrast, the clusters were developed in the optimized feature space.

The clustering in the original expanded feature space gave the best results for the Samusik dataset regarding CHI. Still, it is worth noting that the k-Means results in the binary expanded dataset also produced the second-best results in terms of DBI.

4.4 Results for the drug-resistant Tuberculosis dataset from Stellenbosch University

This section presents the results for the dataset provided by Stellenbosch University partners from studies on drug-resistant Tuberculosis. Since the annotations for the cell types are unknown, the procedure was fully unsupervised. The dataset was applied to the transformations and methods presented in the doctoral thesis. Moreover, the interesting aspect was establishing the cluster composition in terms of the patient cells' counts. To accomplish this, the two-way ANOVA identified the factors significantly influencing cluster formation.

4.4.1 Dataset preparation

The dataset contains over ten million cells and 32 markers (details were presented in *Section 1.5.1.*) The dataset was ArcSinh transformed, and the batch effect was removed with cyCombine.

GMM decomposition was applied to each marker to create the set of components that were further used to get probabilities for each cell belonging to that component (Figure 4.23.). The GMM decomposition of all markers resulted in 314 new features. The binary representation of the expanded feature space was also prepared.

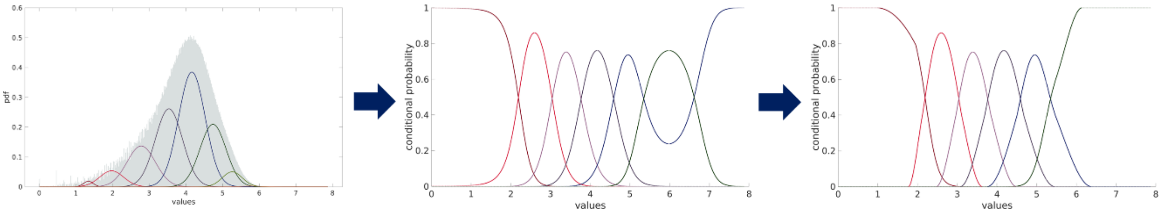


Figure 4.23. Process of creating the expanded feature space for the dataset from scientific partners on a CD45 marker example.
The marker expression values are decomposed into GMM components, and the conditional probability is calculated. The conditional probability lines are further corrected into the membership function. (Source: personal collection).

UMAP projection was generated for the original (Figure 4.24.) and binary expanded domain (Figure 4.25.). Additionally, some isolines are superimposed into the UMAP plot to reveal dense areas of cells. The lighter the color, the more cells the region contains. The visualization aimed to help evaluate the clustering results since the expert's labels are not given. The clusters should more or less match the patterns visible in the UMAP projection. Since the visible peaks are elongated, more groups should be expected in that region because it is probably a mixture of multiple cell types.

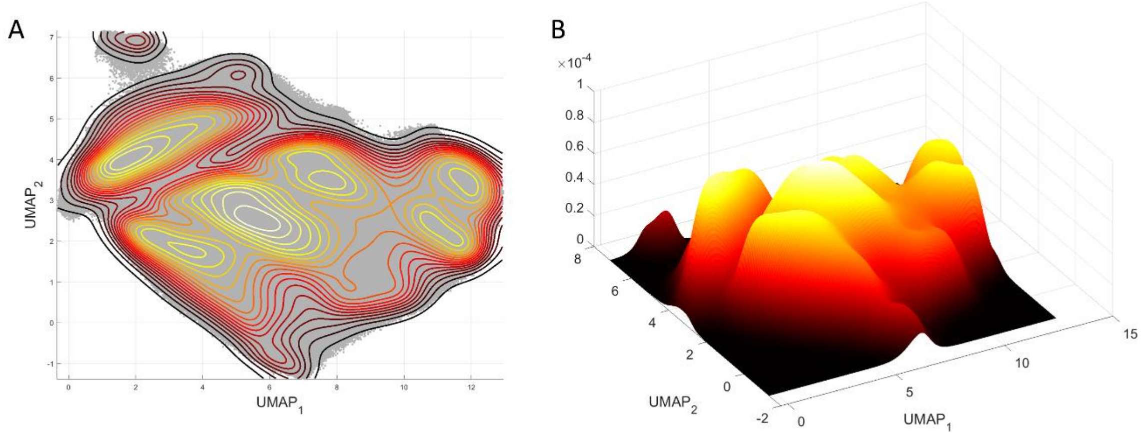


Figure 4.24. UMAP projection for the original expanded domain of the Tuberculosis dataset.
A) UMAP projection with contour plot showing the most densely occupied areas by cells from the dataset. Yellow and white colors indicate the densest areas. B) The 3D surface plot shows the UMAP space's dense areas. (Source: personal collection).

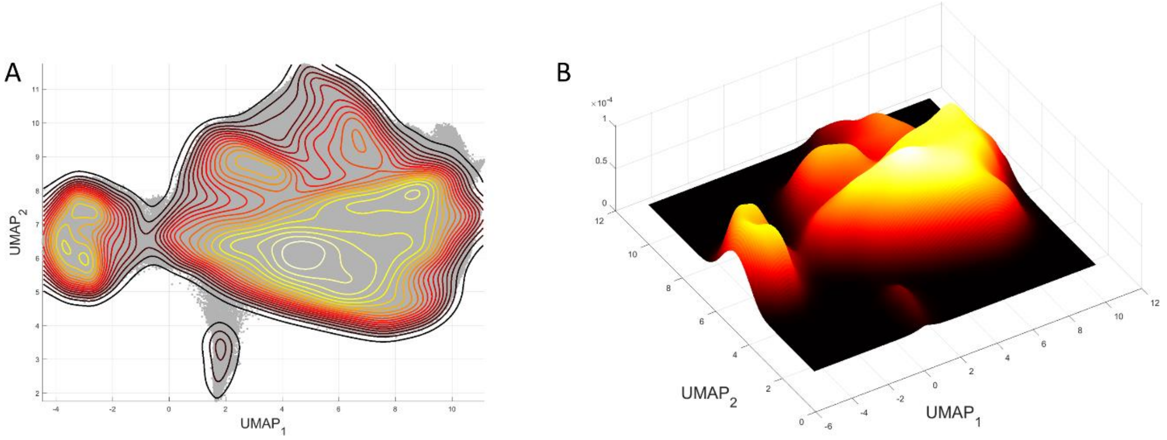


Figure 4.25. UMAP projection for the binary expanded domain of the Tuberculosis dataset.
A) UMAP projection with contour plot showing the most densely occupied areas by cells from the dataset. Yellow and white colors indicate the densest areas. B) The 3D surface plot shows the UMAP space's dense areas. (Source: personal collection).

4.4.2 Clustering results

The cell-type identification with PARC was conducted on the processed dataset following the pipeline invented with Samusik's dataset. First, the algorithm with default parameters and 'jac_weighted_edges' set to False identified the primary cell populations. Then, the second division was applied to the results from the first division to get the final number of subpopulations. The steps were performed in the locally optimized feature space with the thresholds estimated during the analysis of Samusik's dataset.

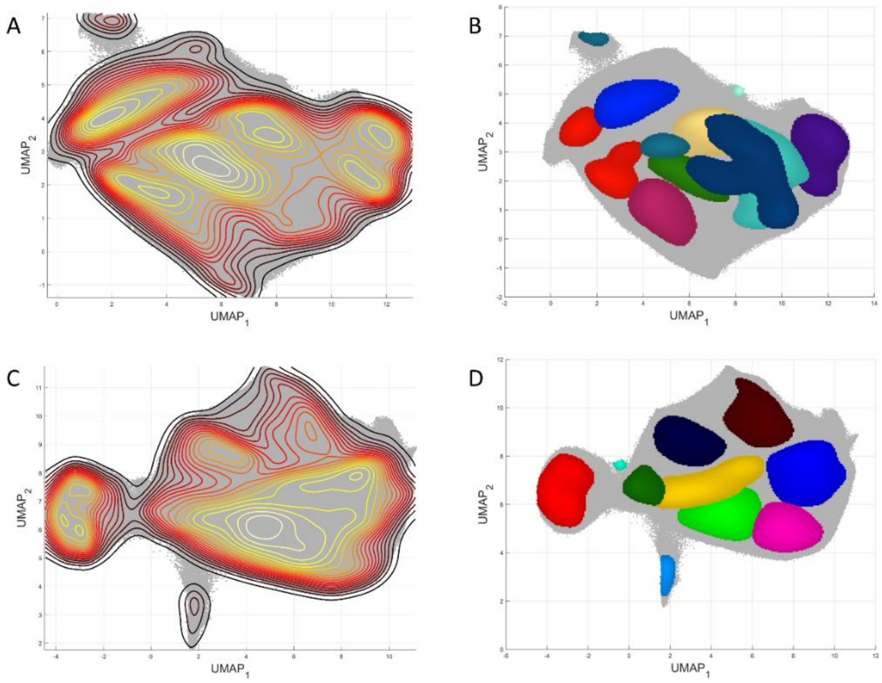


Figure 4.26. Identified primary cell populations in the Tuberculosis dataset after the first division with PARC.
A) UMAP projection for the original expanded domain. B) First division (ten clusters) in the original expanded domain. C) UMAP projection for the binary expanded domain. D) First division (ten clusters) in the binary expanded domain. (Source: personal collection).

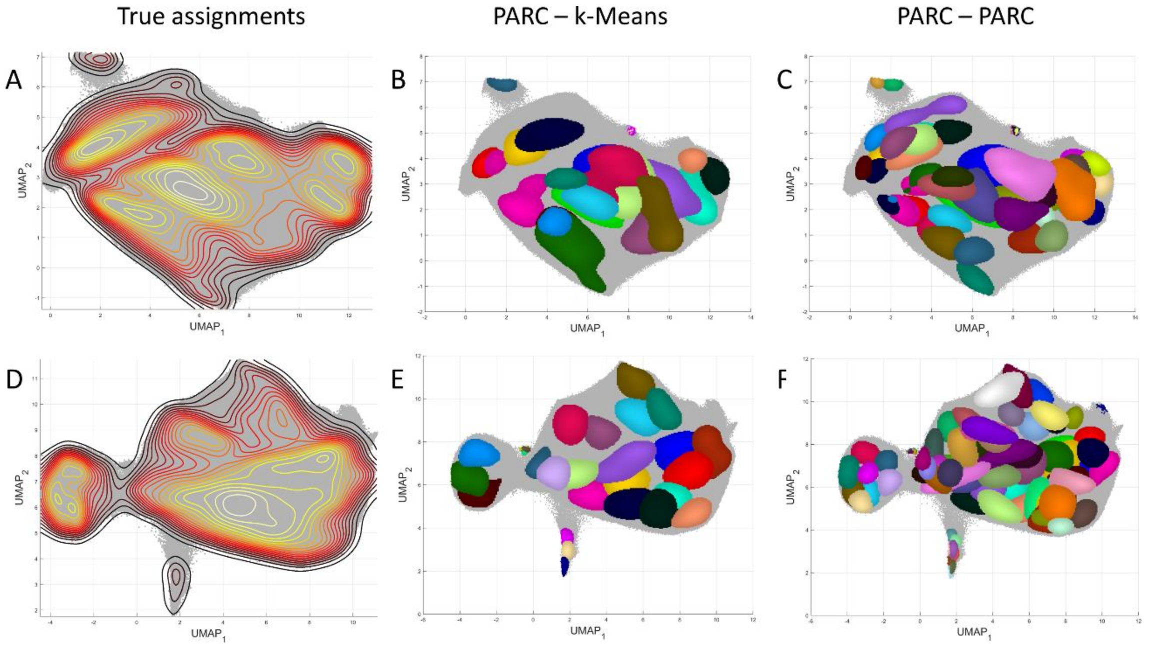


Figure 4.27. Second division results for the Tuberculosis dataset.
A) UMAP projection for the original expanded domain. B) Second division with k-Means algorithm in the original expanded domain. C) Second division with the PARC method in the original expanded domain. D) UMAP projection for the binary expanded domain. E) Second division with k-Means algorithm in the binary expanded domain. F) Second division with the PARC method in the binary expanded domain. (Source: personal collection).

The first division resulted in ten original and binary expanded domain clusters (Figure 4.26). The clusters were further divided with PARC and k-Means algorithms, and the details are presented in Table 4.8. The best approach was combining PARC (first division) with k-Means (second division) in the binary expanded domain, resulting in CHI equal to 279392.04 and DBI 3.8382, with 27 cell populations. The second best approach was PARC and k-Means in the original expanded domain (CHI=184546.25).

Table 4.8. Clustering results for the Tuberculosis dataset.

Transformation	1st division	2nd division	#clusters	CHI	DBI
Original	PARC	k-Means	23	184546.25	7.1780
	PARC	PARC	50	118034.15	8.4905
Binary	PARC	k-Means	27	**279392.04**	**3.8382**
	PARC	PARC	75	137326.07	3.7878

Visually, the best approach generated well-separated clusters consistent with the structure of UMAP projection, including small aggregates of cells and bigger cell populations.

The second division using PARC usually generates more clusters than k-Means with GAP statistics.

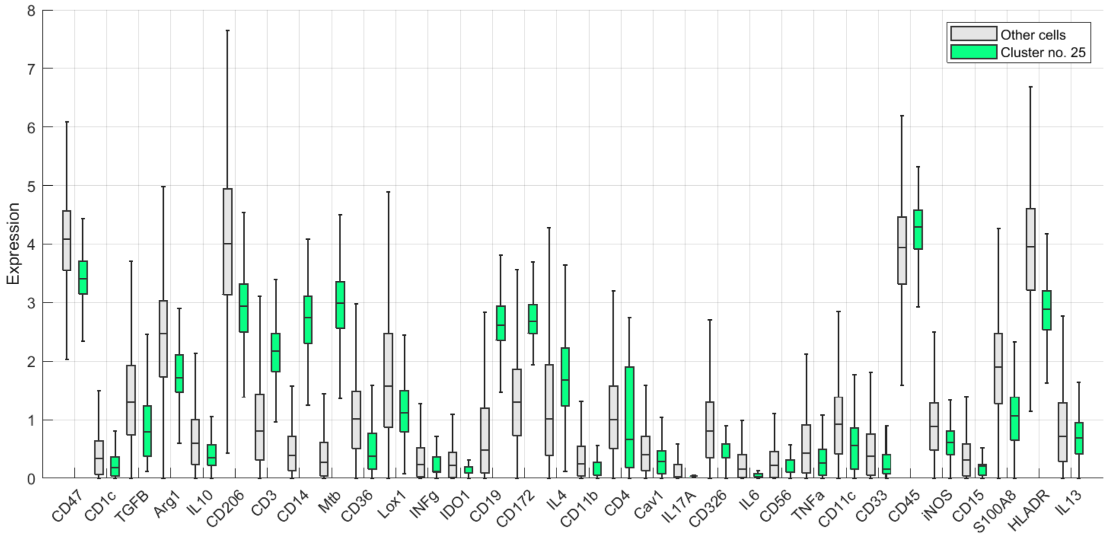


Figure 4.28. Comparison of expression values between the chosen rare subpopulation of cells and others. Gray color indicates all cells except those belonging to cluster number 25, which are presented in green. For some markers, the expression differences between the two groups of cells are significant (CD19, CD172, CD14, Mtb, IL4).

Among the resulting clusters, three contain less than 1% of all data (around 0.005% each) and are considered rare subpopulations. One of the small clusters was compared to other cells to investigate whether its marker profile differs significantly. For this purpose, the small green group was chosen. The original marker expressions were compared between the cluster and other cells in Figure 4.28. The dominant components in the expanded space were also compared with five different clusters placed in various locations in the UMAP space, as indicated in Figure 4.29. The figure presents four, the most relevant for the green cell group, markers for simplicity.

Based on the expression values in Figure 4.28., it can be concluded that the rare subpopulation is indeed distinct from other clusters. The marker profile for that group has significant differences compared to others.

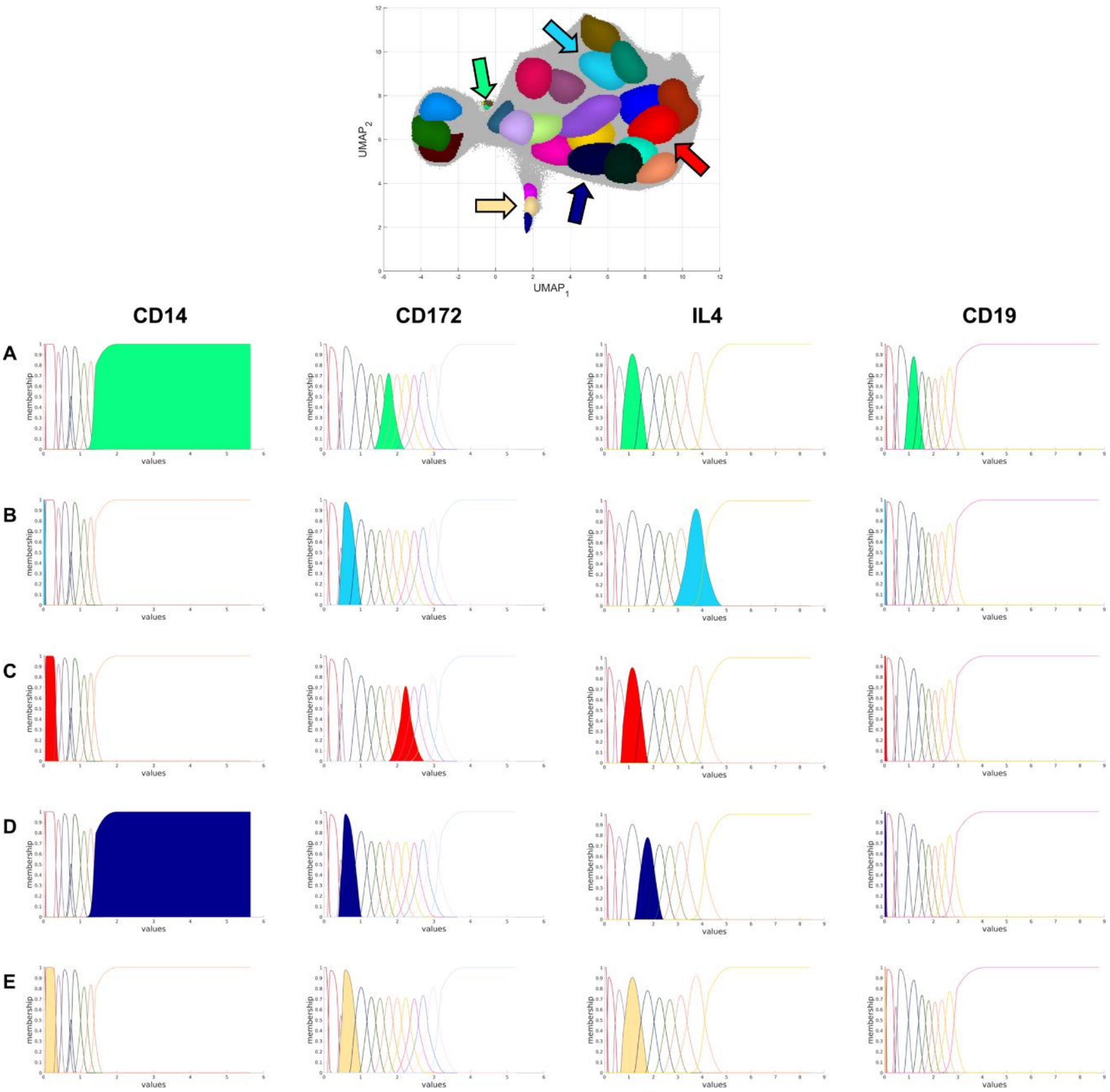


Figure 4.29. Comparison of the dominant marker components for the chosen rare subpopulation and four others.
The visualization presents four markers: CD14, CD172, IL4, and CD19. The color indicates the appropriate cluster. For example, the green one is the rare subpopulation that contains about 0.005% of cells in the Tuberculosis dataset. Other groups are placed in different locations in the UMAP projection.

4.4.3 Cluster composition analysis with two-way ANOVA test

The created clusters were analyzed with a two-way ANOVA test to determine if a group of patients and treatment (stimulation conditions) impact the composition of each cell population. In addition, a Kolmogorov-Smirnov test for normality determined that the patient cell frequencies have a distribution different than Gaussian. Therefore the test was performed on ranks. Results from the ANOVA test for each identified cell population are presented in Table 4.9.

The ANOVA results indicate no interaction between the patient group (Healthy Donors, Other Lung Diseases, and Tuberculosis) and the treatment (Unstimulated, PPD, or PHA stimulated). However, the significant impact (p-value<0.05) that is also confirmed with a large effect size (>0.14) has the patient group on the composition of the cell populations.

Table 4.9. Two-way ANOVA results (p-value, effect size) for each cluster of the Tuberculosis dataset.

No.	**Treatment-adjusted ANOVA group**	**Treatment-adjusted ANOVA group effect size**	**Group-adjusted ANOVA treatment**	**Group-adjusted ANOVA treatment effect size**	**Group-treatment interaction**	**Interaction effect size**
1	0,0082	0,2791	**0,0264**	0,2048	0,4136	0,1037
2	0,0007	0,4544	0,1224	0,1137	0,9931	0,0061
3	0,0000	1,3149	0,7023	0,0183	0,9896	0,0076
4	**0,9574**	0,0022	0,3415	0,0566	0,9656	0,0145
5	**0,0791**	0,1389	0,6309	0,0239	0,9011	0,0268
6	**0,2565**	0,0723	0,8496	0,0084	0,6578	0,0626
7	**0,0994**	0,1257	**0,0275**	0,2024	0,9815	0,0103
8	**0,3758**	0,0515	**0,0370**	0,1841	0,9860	0,0089
9	**0,2765**	0,0681	0,5494	0,0312	0,9950	0,0052
10	**0,1994**	0,0862	0,1540	0,1007	0,9619	0,0154
11	0,0005	0,4792	0,6717	0,0206	0,4574	0,0952
12	**0,6721**	0,0206	0,2882	0,0659	0,8824	0,0298
13	0,0005	0,4778	0,6967	0,0187	0,8753	0,0309
14	0,0244	0,2098	0,2006	0,0859	0,8753	0,0309
15	0,0000	1,1773	0,8720	0,0070	0,5114	0,0856
16	0,0000	1,097	0,7369	0,0158	0,6174	0,0686
17	0,0001	0,5793	0,8472	0,0085	0,9979	0,0033
18	0,0010	0,4277	0,8680	0,0073	0,9954	0,0050
19	0,0004	0,5028	0,8121	0,0107	0,9511	0,0177
20	0,0022	0,3679	**0,0034**	0,3383	0,9680	0,0139
21	0,0300	0,1969	**0,0285**	0,2001	0,9715	0,0131
22	0,0016	0,3890	0,0758	0,1414	0,5682	0,0763
23	0,0283	0,2005	0,9268	0,0039	0,8875	0,0290
24	**0,0627**	0,1526	0,3603	0,0537	0,9623	0,0153
25	**0,4908**	0,0372	0,1019	0,1242	0,9208	0,0234
26	**0,5202**	0,0341	0,1446	0,1043	0,9543	0,0170
27	**0,7715**	0,0134	0,2110	0,0831	0,9119	0,0250

To check whether some of the created clusters contain more cells from a particular group of patients, boxplots of each patient group proportions (a portion of cells from the sample that is present in a given cluster) are presented in Figure 4.30. Moreover, the composition of each cluster is illustrated in Figure 4.31. in the form of a barplot showing the contribution of each group of patients.

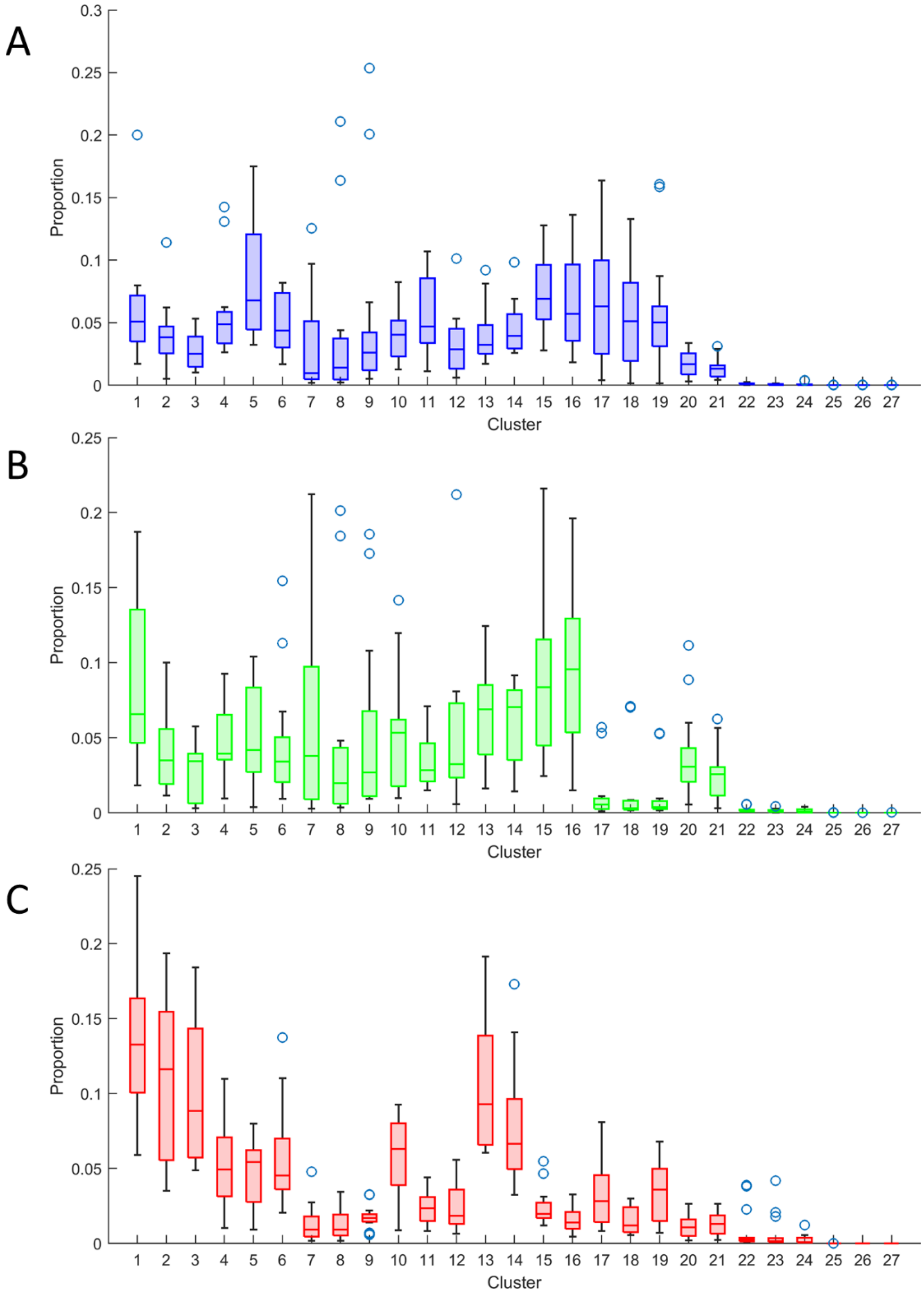


Figure 4.30. The proportion of each patient group in each of the resulting clusters for the Tuberculosis dataset.
A) Healthy Donors group proportions in each cluster. B) Other Lung Diseases group proportions in each cluster. C) Tuberculosis group proportions in each cluster. (Source: personal collection).

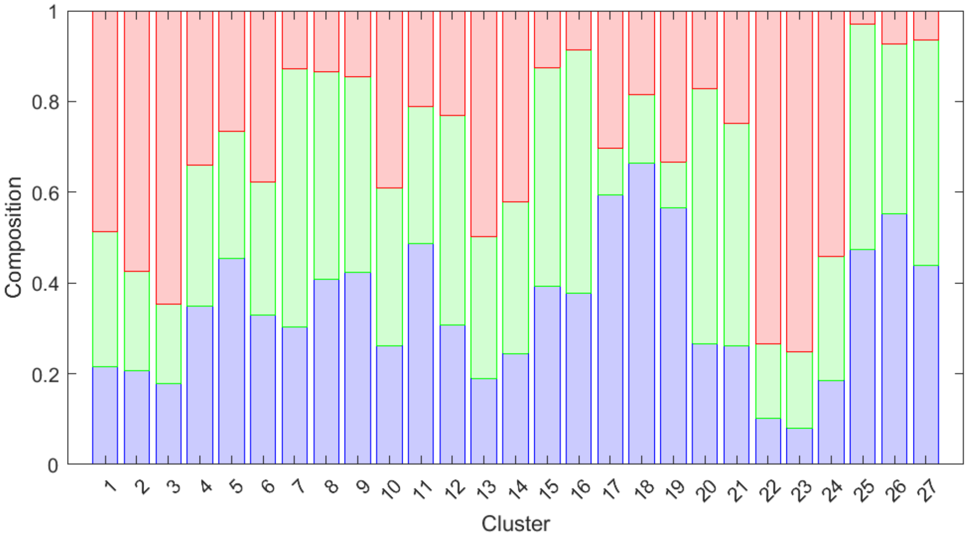


Figure 4.31. Composition of clusters in Tuberculosis dataset.
Red – Tuberculosis (TB) group, green – Other Lung Diseases (OLD) group, and blue – Healthy Donors (HD) group. Each bar shows the contribution of the cells from each group of patients in the cluster. For example, cluster nr 22 contains mostly cells from the Tuberculosis group of patients. (Source: personal collection).

Clusters 22 to 27 contain the smallest number of cells. While 22 to 24 have mostly Tuberculosis cells, 25 to 27 are a mixture of Healthy Donors and OLD cells. Based on Figure 4.31. and ANOVA results, seven clusters are composed of TB cells (1, 2, 3, 13, 14, 22, 23, 24). In addition, about seven groups contain mostly HD cells (11, 17, 18, 19), and about five have more OLD cells than the others (7, 15, 16, 20, 21).

4.5 Discussion and conclusions

Analysis of single-cell mass cytometry data is challenging and requires careful investigation and implementation of methods resulting from the high dimensional feature space and the data heterogeneity. Data heterogeneity may significantly impact the clustering results – algorithms may find more groups of phenotypically similar cells than the number identified by an expert. Therefore it is underlined by authors of many publications in the field [2], [44], [46] that depending on the expert annotations during clustering evaluation may not be the best solution. High-dimensional data makes it difficult to find rare cell populations, especially if the difference between the number of cells in specific and rare cell groups is significant. These challenges are still being investigated, and many researchers are trying to find the optimal solution.

In the doctoral thesis, the problems regarding cell-type identification were examined using Samusik's dataset [22]. Three main aspects were addressed that bring novelty to the topic: an exploration of the data heterogeneity and identification of cell subpopulations that were further included in the evaluation of clustering methods; generation of an expanded domain where each feature brings potential information about rare subpopulations; and proposition of cell identification approach with feature selection and step division in the expanded domain.

Samusik's dataset was preprocessed and embedded into a two-dimensional UMAP space for visualization. UMAP revealed dense areas containing aggregates of similar cells that were further identified as cell subpopulations after 2D GMM decomposition in the UMAP space. Bayes Factor determined the optimal number of components. However, the outliers (single points scattered in the space) greatly influenced the results. The total number of identified subpopulations was 90; each cell type was decomposed into 1-8 components after filtration of the candidates with too low component proportions. The found subpopulations were then included in the calculation of evaluation metrics.

For each resulting cluster, the centroid in the original feature space was computed and compared to each centroid from the set of subpopulation candidates. The subpopulation to which the distance was the smallest determined the new label - the parent cell type label of the subpopulation was assigned as a new label for the cluster. This novel approach makes the results more independent from the expert's annotations, still considering them. That means if the clustering algorithm finds two subpopulations of the same cell type identified by the expert, it shouldn't be "punished" for separating the cluster into two smaller ones since the data heterogeneity is the cause of the decision. Expert annotations are burdened with human errors because the accurate manual annotation of single-cell high-dimensional datasets is impossible. The knowledge of cell subsets and their biological functions is not yet fully understood.

After preprocessing, the expanded domain was created and corrected, increasing from 38 to 318 features. The proposed correction algorithm for conditional probability lines works as intended, discarding inactive components, and fixing the lines to have the right order and only one "peak." However, the proposed algorithm has limitations and disadvantages. It requires the knowledge of Gaussian Mixture Model components. Suppose the dataset has a massive number of observations (like the Tuberculosis dataset from Stellenbosch University with more than 10 million cells). In that case, the Gaussian Mixture Decomposition may take a long time and result in a large number of components. The parameters and criteria used in the algorithms should be adapted to overcome the problem, for example, the minimal allowed sigma for a Gaussian component.

Moreover, normalization may distort the correction and therefore is impossible to apply. The new lines no longer represent conditional probabilities; as a result, the transformation was called a membership function. On the other hand, the lines preserve their original shape at the top of the peaks and the points of intersection of the lines, so it still can be used for determining cut-off values, which is a widely used approach in different fields. Several improvements will be applied to the algorithm for faster and better results, like correcting only those lines requiring this. The correction algorithm was tested on different datasets and is publically available and updated regularly. (https://github.com/Aleksandra795/membership_function.git)

Usually, high-dimensional datasets are applied to downsampling and dimensionality reduction to remove noninformative features, instead of expanding dimensions. Still, the new features carry a subset of information derived from the origin marker expression values. By assumption, such features should be more informative than the origin feature and help to find even rare cell populations.

The unsupervised Calinski-Harabasz Index (CHI) and Davies-Bouldin Index (DBI) were the most important scores for evaluating the results. The indices were calculated using cluster labels returned by a model and the dataset in the regular domain in order to score the model based on the information that is not transformed. The supervised metrics were used to show how their values change when subpopulations are included, but the metrics are not an appropriate indicator of the clustering results since they measure the compatibility of the results with the experts' annotations which may lead to misinterpretation. The unsupervised metrics should be safer because they depend on the data and its hidden structure.

Moreover, the Calinski-Harabasz Index was the leading choice in deciding which method was the best at each analysis step. The higher the values, the better defined and separated, dense, convex clusters – as we could expect from the cell groups (which is partially visible in the UMAP projection). On the other hand, the Davies-Bouldin Index disadvantage is having higher values for convex clusters; therefore, it is possible to obtain better results for Calinski-Harabasz Index and, at the same time, worse for Davies-Bouldin Index. This situation took place a few times during the analysis. If the results were similar, the second factor in deciding which method was better was the number of clusters. Therefore, it was important that the proposed method for cell population identification had good metrics and could find a reasonable number of populations.

Among the proposed methods for cell-type identification so far, only a few have the potential to process big datasets and find rare subpopulations. The study compared three methods using Samusik's dataset in the regular and expanded feature space domain. The clustering of the dataset in the expanded domain resulted in better scores than the regular domain, confirming the dimensionality expansion's usefulness. The best results were obtained for the PARC algorithm (both domains), with the best values of CHI=95594.57 and DBI=1.3168 for the expanded feature domain. Therefore only PARC was used for further comparisons of results.

The expanded feature space may influence the clustering results negatively due to the possible course of dimensionality. The new features carry valuable information that can distinguish various types of cells, but the model may lose this information when the decision about creating a cluster is based on all available features. This may explain the smaller number of identified cell populations using the expanded domain with PARC. Also, algorithms may not work well for data distribution so far from Gaussian distribution. The expanded feature space in the original form gathers the observations on two ends of the feature range: near zero for the cells that do not express the particular feature and near one for those that express the feature. A small part of the dataset is placed somewhere in the middle of the distribution (Figure 4.4.). The binary transformation directly shows if a cell expresses a given feature instead of fuzzy membership and speeds up the computations. Therefore it was decided to check its potential.

Finally, to address the mentioned problems with the impact of dimensionality on model and clustering results, some feature selection criteria were proposed. The tested possibilities were the filtration using mean values, variances, Relative Diversity Index, and the ratio of non-zero (minimal) values to all values. The best method was found empirically after numerous experiments – the Relative Diversity Index gave the best results. However, the other transformations resulted in an overestimated number of clusters that reached even few thousand, which indicated that the algorithm is not performing well and the result is not optimal.

The threshold for feature selection was set to 0.17 based on the GMM decomposition of the RDI distribution values for the original expanded domain and 0.11 for the binary expanded domain. The same thresholds were further used for the Tuberculosis dataset. The cut-off points discarded most features with a negligible impact on the clustering results; however, the approach should be tested in more detail. Furthermore, the same cut-off value may not be optimal for each dataset and algorithm. Therefore, a more data-driven approach should be proposed in the future. Another interesting direction would be using the relative diversity index combined with another feature selection method.

After applying the transformations and feature selection methods, the PARC results were similar for the original and binary expanded feature space. Based on Calinski-Harabasz Index, the original expanded domain was better. On the other hand, the number of clusters was smaller than the results without the feature selection, and the Davies-Bouldin Index was worse than for the binary expanded domain. The number of groups reached 13 and 14 for the original and binary expanded domains, respectively, which may indicate that the populations overall are well-defined, dense, and convex. Still, since it is known that there are more cell types, it was decided to introduce the second division of the clusters.

A few things may explain the results. First, although PARC is fast and one of the most accurate methods, it may not be suitable for the expanded feature space. Secondly, the filtration criterion may not be sufficient to preserve only those features that have a tangible impact on cluster formation. Perhaps a completely different approach would select more informative features than the checked statistics (mean, variance, Relative Diversity Index, and non-zero values to all ratio). Finally, the expanded feature space is a complex concept that has not been used before in such a way. It requires a lot of different kinds of experiments to gain an understanding of the data structure and to be able to use it properly and take full advantage of the information it carries. Nevertheless, the first impression (Figure 4.13.) indicates that expanding the feature space with GMM components is beneficial.

For the second division, it was decided to perform the clustering using PARC and k-Means algorithms on both dataset transformations with the proposed feature selection. Several experiments were conducted to see which approach, dataset, and algorithm would result in the best identification. In the end, a k-Means algorithm gave the best results. Each cluster from the first division was considered for further subdivision with the GAP statistic, which also determined the optimal number of groups making the whole algorithm independent of the prior knowledge about cell types within the dataset.

The results were worse for PARC as the second division algorithm, and the number of clusters was heavily overestimated. On the other hand, PARC can find relevant subpopulations that were overlooked by k-Means. What is essential, because the k-Means algorithm could not process a huge number of cells, sampling was introduced – from each cluster, a sample of 10,000 cells was randomly selected and applied to clustering. Given the cluster centroids, the rest of the cells were assigned to the closest one. Sampling may cause some of the rare subpopulations to be unidentified. In the future, this step should be refined to minimize the possibility of losing rare cell types.

Results from the second division were better than those obtained using the regular space regarding the Calinski-Harabasz Index. Although not all known cell populations were found, and some of the bigger groups of cells were divided into smaller subpopulations, the metrics were calculated using the labels on the dataset in the regular domain. Therefore, the CHI value increase indicates a real separation improvement.

The overlooked known cell populations might be the result of creating the expanded feature space. In the UMAP projection, cell types representing a particular family of cells that are phenotypically similar and perform similar functions are placed closer to each other than in the regular domain. This is visible in Figure 4.16.C-D. Therefore some of the types of cells started to overlap. It can be a UMAP artifact since it is a two-dimensional representation of the high-dimensional dataset, but it also may be a symptom of what is happening with the distances between cells in the expanded space.

Another thing worth noting is that when the division is performed in a locally optimized feature space, the position of the data points in the space is different. Therefore, the UMAP projection for the selected markers would also look different. Returning to the UMAP space generated using the complete set of features could not correctly reflect the subpopulations. As a result, some of them overlap in the visualization.

The proposed algorithms for cell-type identification were successfully applied to a twenty times larger Tuberculosis dataset than the one used during implementation. In addition, the two-step approach enabled labeling cell clusters in a very dense agglomerate of cells.

Interestingly, the best results for the Tuberculosis dataset were obtained in the binary expanded domain. The transformation also worked well for Samusik's dataset, although the original expanded did better. Nevertheless, the results point to a great potential for that kind of transformation that may lead to a state-of-the-art approach when combined with a set of adapted algorithms for processing and clustering binary data. This idea will be further developed in future studies.

The two-step approach successfully identified three rare subpopulations of cells. Investigation of the marker profiles for one of them revealed significant differences compared to other cells. That subtype's dominant marker components are often placed in the middle of the marker's range values. The proposed expanded feature domain allowed for extracting the information during clustering.

The two-way ANOVA test identified cell populations that significantly differ in the number of cells from each patient group. A deeper investigation of the cells and their processes may help understand Tuberculosis patients' drug resistance.

The high density of points in the UMAP projection of the Tuberculosis dataset is independent of the UMAP parameters; therefore, the UMAP uses the same set of parameters as the projections for other datasets. The visualization reveals the difficulty of finding cell subpopulations in such an extensive dataset. There is also the possibility that the challenge is a consequence of combining three groups of patients whose cells were subjected to three other stimulation conditions. Cells may be in various states with varying degrees of response to a given factor, causing them to stick together. Perhaps separating the groups of patients and merging the resulting similar clusters between experiments would be a better approach.

The Tuberculosis dataset provided by partners from Stellenbosch University is complex and should be further investigated. Furthermore, independently from the clustering technique, the other ways of analysis should be checked, like the clustering of HD, OLD, and TB patients separately or separating the samples based on stimulation conditions.

Another interesting future direction could be developing approaches that use deep learning techniques for clustering. Deep learning tries to answer the problems of traditional machine learning techniques. Neural networks can process massive amounts of data, making them suitable for single-cell datasets. The learned data representation is usually more informative and results in better clustering. These solutions comprise three main parts: the feature extractor (usually an Autoencoder or Convolutional Neural Network and their variations), network loss, and the clustering loss (principal loss + auxiliary loss) [59]. The field of deep clustering is still evolving, some architectures have been proposed, but their training is challenging; therefore, developing methods for clustering based on deep learning is an exciting possibility for further analysis.

5 Summary of the doctoral thesis

The doctoral thesis aimed to propose new solutions for mass cytometry data analysis of high-dimensional datasets to be able to process the samples provided by scientific partners from Stellenbosch University. The Tuberculosis dataset consists of more than 10 million cells and 32 markers. A large number of observations is a challenge for most of the existing solutions – some of the methods cannot fit the calculations in computer memory, or the measures take several days. The results are often inaccurate even if an algorithm can process such an extensive dataset. Another significant limitation of the existing solutions is the manual gating performed for filtering unwanted events or identifying cell populations by experts for model evaluation purposes. In the doctoral dissertation, we tried to address the mentioned problems using high-dimensional public datasets. Because the data are usually smaller than the Tuberculosis one, it was faster and easier to implement and test new approaches. In addition, the data provided by partners were unsupervised and did not contain all necessary markers or observations to perform all tasks.

Manual pre-gating, if performed inaccurately, may lead to removing cells and contamination of the data with debris, beads, or double events. Moreover, the manual approach makes it impossible to reproduce the results. More automated techniques appeared to overcome this problem, but the user must still decide about various parameters that may affect the final result. Therefore, a new data-driven pre-gating method was proposed that can be used without predefining any parameters. While trained on one dataset, it is easily generalizable to others and still works well even when some of the pre-gating markers are missing.

Another topic discussed was batch effect correction. There was no robust and accurate method for reducing the technical variance in the data introduced during an experiment for a long time. In recent years some techniques have been proposed. Since there is more than one method to choose from, someone may wonder which of them will fit the best to a given dataset and how the choice may further influence the cell-type identification results. The comparison analysis using a subset of the Tuberculosis dataset with the following clustering allowed granting cyCombine as the superior method over others.

Nevertheless, the essential part of the doctoral dissertation was *Chapter 4*, which focuses on cell-type identification approaches. Some authors use manual assignments of cells into the known types that experts prepared. However, following the predefined types of cells while implementing new clustering techniques may lead to results that are not optimal. Manual annotations lack the consensus between experts on the cell types and their "borders" in feature space. The experts generate the annotations through gating on bi-axial plots of pairs of markers known to be characteristic of certain types of cells. Such gating assumes relationships between pairs of features, ignoring other relations. With the uncovered knowledge about the tissue composition analyzed, the annotations are far from perfect and may lead to misunderstanding of the found groups. The proposed novel approach for model evaluation considers data heterogeneity that allows the model to find additional aggregates of cells. The technique may be used to evaluate clustering model performance with a degree of freedom for the model to decide whether the assumed cell types contain distinct groups of observations that may differ significantly.

As mentioned above, existing solutions are limited to the number of observations – the algorithms are not scalable or accurate enough to generate relevant results. In the study, a new approach for the clustering of cells was proposed. One crucial observation from a series of experiments during the research was separating the information about cell populations from one feature to several new ones after decomposition with Gaussian Mixture Model. Specific rare cell populations highly expressed some of the new features; therefore, it was decided to examine the potential of the expanded dataset domain. The presented results confirm the usefulness of data transformation in cell identification studies, but the nature of new features is complex and needs more investigation to be fully appreciated. The first experiments revealed the restraint of the group division algorithm, which may result from the underlined similarity of the cell type origin in the expanded space; therefore, applying the second division of the created clusters allowed for more in-depth identification.

Nevertheless, the mass cytometry data analysis of high-dimensional datasets is a broad topic that can be investigated for many years. Despite the proposed solutions for many problems the scientific community faces, the algorithms still can be improved in terms of accuracy and computation time. The many possibilities and continuous technological development, including the rapid evolution of artificial intelligence systems, make the topic exciting and worthy of further research. Such an interesting aspect may be the application of deep clustering techniques to the cell-type identification task. Deep clustering combines artificial intelligence power of learning patterns and processing high-dimensional datasets with unsupervised statistical methods [59]. Therefore, in my opinion, it is a future direction for mass cytometry and other single-cell technologies.

6 Supplementary materials

Table 6.1. The panel used for staining the Tuberculosis dataset from Stellenbosch University.

Antibody target	Isotope label	Cell types
	Extracellular Antibodies	
CD3	Er170	T-cells
CD14	Eu151	Monocytes/macrophages, granulocytes
CD172	Lu175	Dendritic cells, monocytes/macrophages
CD19	Ho165	B-cells, dendritic cells
CD33	Tm169	Dendritic cells, monocytes/macrophages, granulocytes
CD45	Y89	Leukocytes
CD326	Pr141	Epithelial cells
CD11b	Nd144	T-cells, B-cells, dendritic cells, NK cells, granulocytes, monocytes/macrophages
CD4	Nd145	CD4 T-cells
CD36	Gd155	Dendritic cells, monocytes/macrophages
CD56	Sm149	NK cells, T-cells
Cav-1	Nd146	Ubiquitous expression
Lox-1	Gd156	B-cells, dendritic cells, macrophages, MDSC
CD15	Yb172	Monocytes/macrophages, granulocytes
CD206	Er168	Alveolar macrophages
HLA-DR	Yb174	Leukocytes
CD11c	Tb159	T-cells, B-cells, dendritic cells, NK cells, granulocytes, monocytes/macrophages
CD1c	Dy161	T-cells, B-cells, dendritic cells,

		monocytes/macrophages
CD47	Bi209	Ubiquitous expression
	Intracellular Antibodies	
INF-g	Gd158	
IL-17A	Nd148	
IL-4	Nd142	
IL-10	Er166	
IL-1b	Yb176	
IL-6	Sm147	
iNOS	Yb171	
Arg-1	Dy164	
TGF-b	Dy163	
IDO-1	Gd160	
S100AB	Yb173	
TNF-a	Sm152	
Mtb (PPD)	Eu153	

References

[1] G.-C. Yuan *et al.*, "Challenges and emerging directions in single-cell analysis," *Genome Biol*, vol. 18, no. 1, p. 84, Dec. 2017, doi: 10.1186/s13059-017-1218-y.

[2] L. R. Olsen, M. D. Leipold, C. B. Pedersen, and H. T. Maecker, "The anatomy of single cell mass cytometry data," *Cytometry Part A*, vol. 95, no. 2, pp. 156–172, Feb. 2019, doi: 10.1002/cyto.a.23621.

[3] X. Liu *et al.*, "A comparison framework and guideline of clustering methods for mass cytometry data," *Genome Biol*, vol. 20, no. 1, p. 297, Dec. 2019, doi: 10.1186/s13059-019-1917-7.

[4] Geneva: World Health Organization, "Global tuberculosis report. Licence: CC BY-NC-SA 3.0 IGO," 2022.

[5] Centers for Disease Control and Prevention, "Tuberculosis (TB), National Center for HIV, Viral Hepatitis, STD, and TB Prevention, www.cdc.gov/tb/worldtbday/history.htm. Accessed: 26 Feb. 2023.".

[6] M. Nowicka *et al.*, "CyTOF workflow: differential discovery in high-throughput high-dimensional cytometry datasets," *F1000Res*, vol. 6, p. 748, May 2019, doi: 10.12688/f1000research.11622.3.

[7] A. I. Abdelrahman *et al.*, "Metal-containing polystyrene beads as standards for mass cytometry," *J Anal At Spectrom*, vol. 25, no. 3, p. 260, 2010, doi: 10.1039/b921770c.

[8] B. H. Lee and A. H. Rahman, "Acquisition, Processing, and Quality Control of Mass Cytometry Data," 2019, pp. 13–31. doi: 10.1007/978-1-4939-9454-0_2.

[9] R. Finck *et al.*, "Normalization of mass cytometry data with bead standards," *Cytometry Part A*, vol. 83A, no. 5, pp. 483–494, May 2013, doi: 10.1002/cyto.a.22271.

[10] C. B. Pedersen and L. R. Olsen, "Analysis of Mass Cytometry Data," 2019, pp. 267–279. doi: 10.1007/978-1-4939-9454-0_17.

[11] J. Spidlen, W. Moore, and R. R. Brinkman, "ISAC's Gating-ML 2.0 data exchange standard for gating description," *Cytometry Part A*, vol. 87, no. 7, pp. 683–687, Jul. 2015, doi: 10.1002/cyto.a.22690.

[12] M. Trussart, C. E. Teh, T. Tan, L. Leong, D. H. Gray, and T. P. Speed, "Removing unwanted variation with CytofRUV to integrate multiple CyTOF datasets," *Elife*, vol. 9, Sep. 2020, doi: 10.7554/eLife.59630.

[13] C. B. Pedersen *et al.*, "cyCombine allows for robust integration of single-cell cytometry datasets within and across technologies," *Nat Commun*, vol. 13, no. 1, p. 1698, Dec. 2022, doi: 10.1038/s41467-022-29383-5.

[14] L. van der Maaten, "Accelerating t-SNE using tree-based algorithms," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3221–3245, 2014.

[15] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction," Feb. 2018.

[16] M. A. ben HajKacem, C.-E. ben N'Cir, and N. Essoussi, "Overview of Scalable Partitional Methods for Big Data Clustering," 2019, pp. 1–23. doi: 10.1007/978-3-319-97864-2_1.

[17] M. Ahmed, R. Seraj, and S. M. S. Islam, "The k-means Algorithm: A Comprehensive Survey and Performance Evaluation," *Electronics (Basel)*, vol. 9, no. 8, p. 1295, Aug. 2020, doi: 10.3390/electronics9081295.

[18] M. R. Karim *et al.*, "Deep learning-based clustering approaches for bioinformatics," *Brief Bioinform*, vol. 22, no. 1, pp. 393–415, Jan. 2021, doi: 10.1093/bib/bbz170.

[19] A. Suwalska, N. du Plessis-Burger, G. van der Spuy, and J. Polanska, "Comparison of Batch Effect Removal Methods for High Dimensional Mass Cytometry Data," 2022, pp. 399–410. doi: 10.1007/978-3-031-07802-6_34.

[20] M. Leipold and H. Maecker, "Phenotyping of Live Human PBMC using CyTOFTM Mass Cytometry," *Bio Protoc*, vol. 5, no. 2, 2015, doi: 10.21769/BioProtoc.1382.

[21] Y. Simoni *et al.*, "Human Innate Lymphoid Cell Subsets Possess Tissue-Type Based Heterogeneity in Phenotype and Frequency," *Immunity*, vol. 46, no. 1, pp. 148–161, Jan. 2017, doi: 10.1016/j.immuni.2016.11.005.

[22] N. Samusik, Z. Good, M. H. Spitzer, K. L. Davis, and G. P. Nolan, "Automated mapping of phenotype space with single-cell data," *Nat Methods*, vol. 13, no. 6, pp. 493–496, Jun. 2016, doi: 10.1038/nmeth.3863.

[23] J. Spidlen, W. Moore, and R. R. Brinkman, "ISAC's Gating-ML 2.0 data exchange standard for gating description," *Cytometry Part A*, vol. 87, no. 7, pp. 683–687, Jul. 2015, doi: 10.1002/cyto.a.22690.

[24] A. Suwalska and J. Polanska, "Preliminary study for a fully automated pre-gating method for high-dimensional mass cytometry data," in *2021 IEEE 21st International Conference on Bioinformatics and Bioengineering (BIBE)*, Oct. 2021, pp. 1–5. doi: 10.1109/BIBE52308.2021.9635492.

[25] G. Finak *et al.*, "OpenCyto: An Open Source Infrastructure for Scalable, Robust, Reproducible, and Automated, End-to-End Flow Cytometry Data Analysis," *PLoS Comput Biol*, vol. 10, no. 8, p. e1003806, Aug. 2014, doi: 10.1371/journal.pcbi.1003806.

[26] S. M. Castillo-Hair, J. T. Sexton, B. P. Landry, E. J. Olson, O. A. Igoshin, and J. J. Tabor, "FlowCal: A User-Friendly, Open Source Software Tool for Automatically Converting Flow Cytometry Data from Arbitrary to Calibrated Units," *ACS Synth Biol*, vol. 5, no. 7, pp. 774–780, Jul. 2016, doi: 10.1021/acssynbio.5b00284.

[27] H. Chen, M. C. Lau, M. T. Wong, E. W. Newell, M. Poidinger, and J. Chen, "Cytofkit: A Bioconductor Package for an Integrated Mass Cytometry Data Analysis Pipeline," *PLoS Comput Biol*, vol. 12, no. 9, p. e1005112, Sep. 2016, doi: 10.1371/journal.pcbi.1005112.

[28] N. Aghaeepour *et al.*, "GateFinder: projection-based gating strategy optimization for flow and mass cytometry," *Bioinformatics*, vol. 34, no. 23, pp. 4131–4133, Dec. 2018, doi: 10.1093/bioinformatics/bty430.

[29] X. Yang and P. Qiu, "Automatically generate two-dimensional gating hierarchy from clustered cytometry data," *Cytometry Part A*, vol. 93, no. 10, pp. 1039–1050, Oct. 2018, doi: 10.1002/cyto.a.23577.

[30] E. Becht *et al.*, "Reverse-engineering flow-cytometry gating strategies for phenotypic labelling and high-performance cell sorting," *Bioinformatics*, vol. 35, no. 2, pp. 301–308, Jan. 2019, doi: 10.1093/bioinformatics/bty491.

[31] K. R. Atkuri, J. C. Stevens, and H. Neubert, "Mass Cytometry: A Highly Multiplexed Single-Cell Technology for Advancing Drug Development," *Drug Metabolism and Disposition*, vol. 43, no. 2, pp. 227–233, Feb. 2015, doi: 10.1124/dmd.114.060798.

[32] M. Marczyk, "Mixture Modeling of 2-D Gel Electrophoresis Spots Enhances the Performance of Spot Detection," *IEEE Trans Nanobioscience*, vol. 16, no. 2, pp. 91–99, Mar. 2017, doi: 10.1109/TNB.2017.2676725.

[33] K. Lo, R. R. Brinkman, and R. Gottardo, "Automated gating of flow cytometry data via robust model-based clustering," *Cytometry Part A*, vol. 73A, no. 4, pp. 321–332, Apr. 2008, doi: 10.1002/cyto.a.20531.

[34] R. P. Schuyler *et al.*, "Minimizing Batch Effects in Mass Cytometry Data," *Front Immunol*, vol. 10, Oct. 2019, doi: 10.3389/fimmu.2019.02367.

[35] S. van Gassen, B. Gaudilliere, M. S. Angst, Y. Saeys, and N. Aghaeepour, "CytoNorm: A Normalization Algorithm for Cytometry Data," *Cytometry Part A*, vol. 97, no. 3, pp. 268–278, Mar. 2020, doi: 10.1002/cyto.a.23904.

[36] S. Van Gassen *et al.*, "FlowSOM: Using self-organizing maps for visualization and interpretation of cytometry data," *Cytometry Part A*, vol. 87, no. 7, pp. 636–645, Jul. 2015, doi: 10.1002/cyto.a.22625.

[37] M. Ogishi *et al.*, "Multibatch Cytometry Data Integration for Optimal Immunophenotyping," *The Journal of Immunology*, vol. 206, no. 1, pp. 206–213, Jan. 2021, doi: 10.4049/jimmunol.2000854.

[38] I. Korsunsky *et al.*, "Fast, sensitive and accurate integration of single-cell data with Harmony," *Nat Methods*, vol. 16, no. 12, pp. 1289–1296, Dec. 2019, doi: 10.1038/s41592-019-0619-0.

[39] W. E. Johnson, C. Li, and A. Rabinovic, "Adjusting batch effects in microarray expression data using empirical Bayes methods," *Biostatistics*, vol. 8, no. 1, pp. 118–127, Jan. 2007, doi: 10.1093/biostatistics/kxj037.

[40] S. V Stassen, D. M. D. Siu, K. C. M. Lee, J. W. K. Ho, H. K. H. So, and K. K. Tsia, "PARC: ultrafast and accurate clustering of phenotypic data of millions of single cells," *Bioinformatics*, vol. 36, no. 9, pp. 2778–2786, May 2020, doi: 10.1093/bioinformatics/btaa042.

[41] A. Suwalska, M. Socha, W. Prazuch, J. Tobiasz, J. Polanska, and M. Marczyk, "nUMAP: How can we overcome the problem in the visualization in multi dataset comparative studies?," in *red. Computational Oncology and Personalized Medicine - the Challenges of the Future. COPM2022 book of abstracts*, 2022, p. 18.

[42] J. H. Levine *et al.*, "Data-Driven Phenotypic Dissection of AML Reveals Progenitor-like Cells that Correlate with Prognosis," *Cell*, vol. 162, no. 1, pp. 184–197, Jul. 2015, doi: 10.1016/j.cell.2015.05.047.

[43] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science (1979)*, vol. 344, no. 6191, pp. 1492–1496, Jun. 2014, doi: 10.1126/science.1242072.

[44] P. Qiu *et al.*, "Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE," *Nat Biotechnol*, vol. 29, no. 10, pp. 886–891, Oct. 2011, doi: 10.1038/nbt.1991.

[45] B. Becher *et al.*, "High-dimensional analysis of the murine myeloid cell system," *Nat Immunol*, vol. 15, no. 12, pp. 1181–1189, Dec. 2014, doi: 10.1038/ni.3006.

[46] T. Sörensen, S. Baumgart, P. Durek, A. Grützkau, and T. Häupl, "immunoClust - An automated analysis pipeline for the identification of immunophenotypic signatures in high-dimensional cytometric datasets," *Cytometry Part A*, vol. 87, no. 7, pp. 603–615, Jul. 2015, doi: 10.1002/cyto.a.22626.

[47] Y. H. Li *et al.*, "Scalable multi-sample single-cell data analysis by Partition-Assisted Clustering and Multiple Alignments of Networks," *PLoS Comput Biol*, vol. 13, no. 12, p. e1005875, Dec. 2017, doi: 10.1371/journal.pcbi.1005875.

[48] M. H. Spitzer and G. P. Nolan, "Mass Cytometry: Single Cells, Many Features," *Cell*, vol. 165, no. 4, pp. 780–791, May 2016, doi: 10.1016/j.cell.2016.04.019.

[49] Y. Jeong, A. P. Harris, O. Ali, and Y. Jung, "Bayes factor: A useful tool to quantitatively evaluate and compare performance of multiple stature estimation equations," *Forensic Sci Int*, vol. 312, p. 110299, Jul. 2020, doi: 10.1016/j.forsciint.2020.110299.

[50] A. A. Neath and J. E. Cavanaugh, "The Bayesian information criterion: background, derivation, and applications," *WIREs Computational Statistics*, vol. 4, no. 2, pp. 199–203, Mar. 2012, doi: 10.1002/wics.199.

[51] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *J R Stat Soc Series B Stat Methodol*, vol. 63, no. 2, pp. 411–423, 2001, doi: 10.1111/1467-9868.00293.

[52] T. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Commun Stat Theory Methods*, vol. 3, no. 1, pp. 1–27, 1974, doi: 10.1080/03610927408827101.

[53] D. L. Davies and D. W. Bouldin, "A Cluster Separation Measure," *IEEE Trans Pattern Anal Mach Intell*, vol. PAMI-1, no. 2, pp. 224–227, Apr. 1979, doi: 10.1109/TPAMI.1979.4766909.

[54] L. Hubert and P. Arabie, "Comparing partitions," *J Classif*, vol. 2, no. 1, pp. 193–218, Dec. 1985, doi: 10.1007/BF01908075.

[55] D. Steinley, "Properties of the Hubert-Arable Adjusted Rand Index.," *Psychol Methods*, vol. 9, no. 3, pp. 386–396, 2004, doi: 10.1037/1082-989X.9.3.386.

[56] N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison," in *Proceedings of the 26th Annual International Conference on Machine Learning*, Jun. 2009, pp. 1073–1080. doi: 10.1145/1553374.1553511.

[57] A. Rosenberg and J. Hirschberg, "V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 2007, pp. 410–420.

[58] E. B. Fowlkes and C. L. Mallows, "A Method for Comparing Two Hierarchical Clusterings," *J Am Stat Assoc*, vol. 78, no. 383, p. 553, Sep. 1983, doi: 10.2307/2288117.

[59] M. R. Karim *et al.*, "Deep learning-based clustering approaches for bioinformatics," *Brief Bioinform*, vol. 22, no. 1, pp. 393–415, Jan. 2021, doi: 10.1093/bib/bbz170.

Tables

Figures

Abbreviations

AMI – Adjusted Mutual Information

ARI – Adjusted Rand Index

BIC – Bayesian Information Criterion

CHI – Calinski-Harabasz Index

CyTOF – Cytometry by Time-of-flight; mass cytometry

DBI – Davies-Bouldin Index

FCS – Flow Cytometry Standard

GMM – Gaussian Mixture Model

HD – Healthy Donor

mISO – median isoline

NZR – non-zero values ratio

OLD – Other Lung Diseases

RDI – Relative Diversity Index

TB - Tuberculosis

TOF – Time of Flight

Funding

- The European Union scholarship funded the doctoral dissertation through the European Social Fund (grant POWR.03.05.00-00-Z305).
- Joanna Polanska was financed by 02/070/BK_21/0019 project.
- Calculations were carried out using GeCONiI infrastructure funded by NCBiR project no. POIG.02.03.01-24-099/13.