Robust integration of single-cell cytometry datasets

Christina Bligaard Pedersen[1,2,^], Søren Helweg Dam[1,^], Mike Bogetofte Barnkob[3], Michael D. Leipold[4], Noelia Purroy[5,6], Laura Z. Rassenti[7], Thomas J. Kipps[7], Jennifer Nguyen[8], James Arthur Lederer[8], Satyen Harish Gohil[5,9,10], Catherine J. Wu[5,11], and Lars Rønn Olsen[1,2,*]

1. Department of Health Technology, Technical University of Denmark, Kongens Lyngby, Denmark
2. Center for Genomic Medicine, Rigshospitalet - Copenhagen University Hospital, Copenhagen, Denmark
3. Centre for Cellular Immunotherapy of Haematological Cancer Odense (CITCO), Department of Clinical Immunology, Odense University Hospital, University of Southern Denmark, Odense, Denmark.
4. Human Immune Monitoring Center, Institute for Immunity, Transplantation, and Infection, Stanford University School of Medicine, Stanford, CA, USA
5. Department of Medical Oncology, Dana-Farber Cancer Institute, Boston, MA, USA
6. AstraZeneca, Waltham, MA, USA (current employee)
7. Division of Hematology-Oncology, Department of Medicine, Moores Cancer Center, University of California, San Diego, La Jolla, CA, USA
8. Department of Surgery, Brigham and Women's Hospital, Harvard Medical School, Boston, MA, USA
9. Department of Academic Haematology, University College London, London, United Kingdom
10. Department of Haematology, University College London Hospitals NHS Trust, London, UK
11. Broad Institute of MIT and Harvard, Cambridge, MA, USA
^. These authors contributed equally
*. Corresponding author

Contact information
Department of Health Technology, Technical University of Denmark, Kemitorvet, Building 204, 2800 Kongens Lyngby, Denmark
lronn@dtu.dk

Funding
This work was funded by the Independent Research Fund Denmark (grant 8048-00078B to LRO) and the National Institutes of Health (grants P30AR070253 and U01AI138318 to JAL; P01CA206978 and UG1 CA233338 to CJW). SHG was supported by the Kay Kendall Leukaemia Fund.

Conflict of interest
CJW holds equity in BioNTech, Inc; and receives research funding from Pharmacyclics. The authors declare no conflict of interest.

Abstract

Combining single-cell cytometry datasets increases the analytical flexibility and the statistical power of data analyses. However, in many cases the full potential of co-analyses is not reached due to technical variance between data from different experimental batches. Here, we present cyCombine, a method to robustly integrate cytometry data from different batches, experiments, or even different experimental techniques, such as CITE-seq, flow cytometry, and mass cytometry. We demonstrate that cyCombine maintains the biological variance and the structure of the data, while minimizing the technical variance between datasets. cyCombine does not require technical replicates across datasets, and computation time scales linearly with the number of cells, allowing for integration of massive datasets. Robust, accurate, and scalable integration of cytometry data enables integration of multiple datasets for primary data analyses and the validation of results using public datasets.

Introduction

Whether trying to elucidate mechanisms and pathways of diseased cells or characterizing the immune response against infectious diseases or cancer, there is an increasing demand for methods that enable broader and deeper single-cell profiling. Protein expression-based single-cell cytometry has evolved immensely over the past decades. While flow cytometry remains a staple of both basic cell biology research and clinical diagnostics[1], the introduction of mass cytometry (CyTOF) in 2009 increased the number of simultaneously measured markers to 45[2] as issues with signal spillover between reporter molecules and autofluorescence of cells were minimized[3,4]. More recently, spectral flow cytometry enables the measurement of 40 features or more without compromising throughput[5]. Sequence barcoding-based cytometry, such as CITE-seq, has even further increased the number of markers to the hundreds by almost completely eliminating signal spillover[6], and single-cell mass spectrometry is promising to increase feature counts even further[7–9].

Common to all these technologies is the desire to integrate data from different experiments, whether seeking to validate results using external datasets or aiming to increase the breadth and/or depth of the dataset used for a given study. This is rarely directly possible due to technical variance arising from data being generated with different antibody panels, reagent lots, or instruments; at different times; by different operators; etc.[10]. The resulting technical variance is commonly referred to as batch effects, and while many proposed methods offer means to alleviate the problem, robust, flexible, and accurate batch correction of single-cell cytometry data has remained a major unsolved challenge.

Results

The cyCombine batch correction module

To overcome these challenges, we have developed the cyCombine method for integration of cytometry data. The main engine of the cyCombine batch correction module is the tried and true empirical Bayes method for removal of batch effects, ComBat[11]. ComBat was first introduced in

2007 as a tool to address batch effects in DNA microarray data, but the empirical Bayes model has since proven useful for different types of bulk expression data. However, ComBat is not directly applicable to single-cell data, as it is designed to detect and remove technical variance between samples from different batches, while preserving biological variance between samples belonging to homogeneous conditions. However, in single-cell cytometry data, each sample is often characterized by vast heterogeneity in the expression patterns of the different cell types, thus prohibiting explicit modelling of technical and biological variance between samples.

In the cyCombine batch correction module, we address the intra-sample heterogeneity by considering each cell as its own sample and minimize the batch effects for groups of similar cells, one group at a time. The grouping of similar cells is done using a self-organizing map (SOM)[12], with an 8x8 node grid. This means that the cells will initially be clustered into 64 categories. This will typically be enough to capture the diversity of peripheral blood mononuclear cells, but the grid size can be adjusted if less or greater heterogeneity is anticipated. In order to ensure that phenotypically similar cells cluster together across different batches, the expression of each marker is initially standardized within each batch. This is done either by transforming the expression values to Z-scores, which works well for fairly low-variance batches (e.g. data from different batches in an experiment), or ranks, which works well for high-variance batches (e.g. data stemming from different experiments or technologies). The transformed data are then used to cluster the cells using the SOM, and the node labels are assigned to the original expression value cells (**Figure 1a**).

The cyCombine panel merging module

To integrate data from experiments designed with multiple panels of antibodies for increased feature breadth, cyCombine includes a module for panel integration. This module is likewise based on SOM clustering of cells from the different panels using the overlapping markers, followed by probability-based imputation of missing channels by drawing expression values from multi-dimensional kernel density estimates calculated on the cells from the opposing panel (**Figure 1b**). The clustering and multidimensional draws ensure that co-expression patterns and frequencies of subtypes are maintained and only “true” cell types are imputed (see **Supplementary Discussion**).

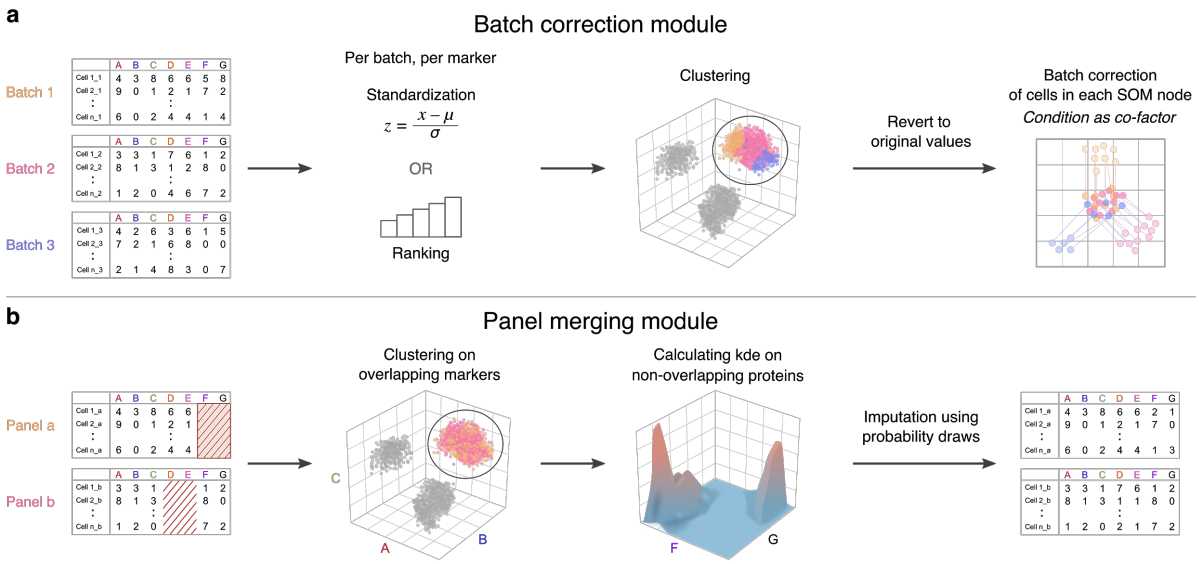

Figure 1. **a**: Batch correction workflow. First, expression values are transformed in each batch to enable co-clustering of samples from all batches. After clustering, the transformed values are reverted to expression values and ComBat is applied to each cluster. **b**: Panel merging workflow. Clustering is performed on overlapping markers, and the missing values for each cell in a panel are imputed using probability draws from the co-clustered cells of the other panel.

cyCombine enables large-scale integration of multi-batch, multi-panel cytometry data

In order to demonstrate that cyCombine enables co-analysis of data from different experimental batches, we generated a CyTOF dataset consisting of 128 samples, run in 7 batches. The experiment contained two conditions: 20 healthy donor samples and 108 chronic lymphocytic leukemia (CLL) samples, collected from 56 patients at two different time points. Each sample was split in two and stained with two different antibody panels, overlapping by 15 markers and differing by 40 markers (**Table 1**).

First, batch effects were minimized in each panel, after which batch effects of the 15 overlapping markers between the two panels were minimized (**Figure 2a-b** and **Supplementary Figure 1**). Then, the two panels were merged by imputing expression data from the non-overlapping markers. The integrated dataset consisted of 12,858,678 cells and the expression of 55 markers. The combined dataset was clustered based on a subset of 23 lineage markers using SOM[12] and ConsensusCusterPlus[13] to 45 meta-clusters, which were labeled manually, cleaned-up, and merged into a total of 29 clusters (**Figure 2c** and **Supplementary Figure 2**). The percentage of cells from each sample assigned to each cluster correlated very strongly (Pearson correlation coefficient = 0.9996) between cells derived from the two panels. For both of the two panels, the batch correction resulted in an earth mover's distance (EMD) reduction of 0.66. Biological variance was retained in both panels, as indicated by the median absolute deviation (MAD) score between pre and post batch correction samples being 0.02 for both panels.

Within the 29 clusters we identified a range of T, NKT, myeloid, and NK cells populations (**Figure 2c** and **Supplementary Figure 3**). Interestingly, we observed that the proportion of the T and

NKT cell compartment was increased in CLL patients (**Figure 2d**), as were circulating stem cells (as identified by CD34+ expression), especially closer to treatment (**Figure 2e**), suggesting marrow stress with higher disease burden. In keeping with previously published data[14–16], we saw a decrease in naive CD8+ T cells, with corresponding increase in the CD8+ TEMRA population when comparing close-to-treatment CLL samples to HDs (**Supplementary Figure 3**). The use of HLA-DR in the staining further identified groups of CD8+ and CD4+ effector memory T cells that increased between CLL time point 1 and 2 with the CD4+ cluster being specifically enriched for PD-1 (**Figure 2f-g** and **Supplementary Figure 3**), similar to that reported by Elston et al. (2020)[15]. See also **Supplementary Discussion**.

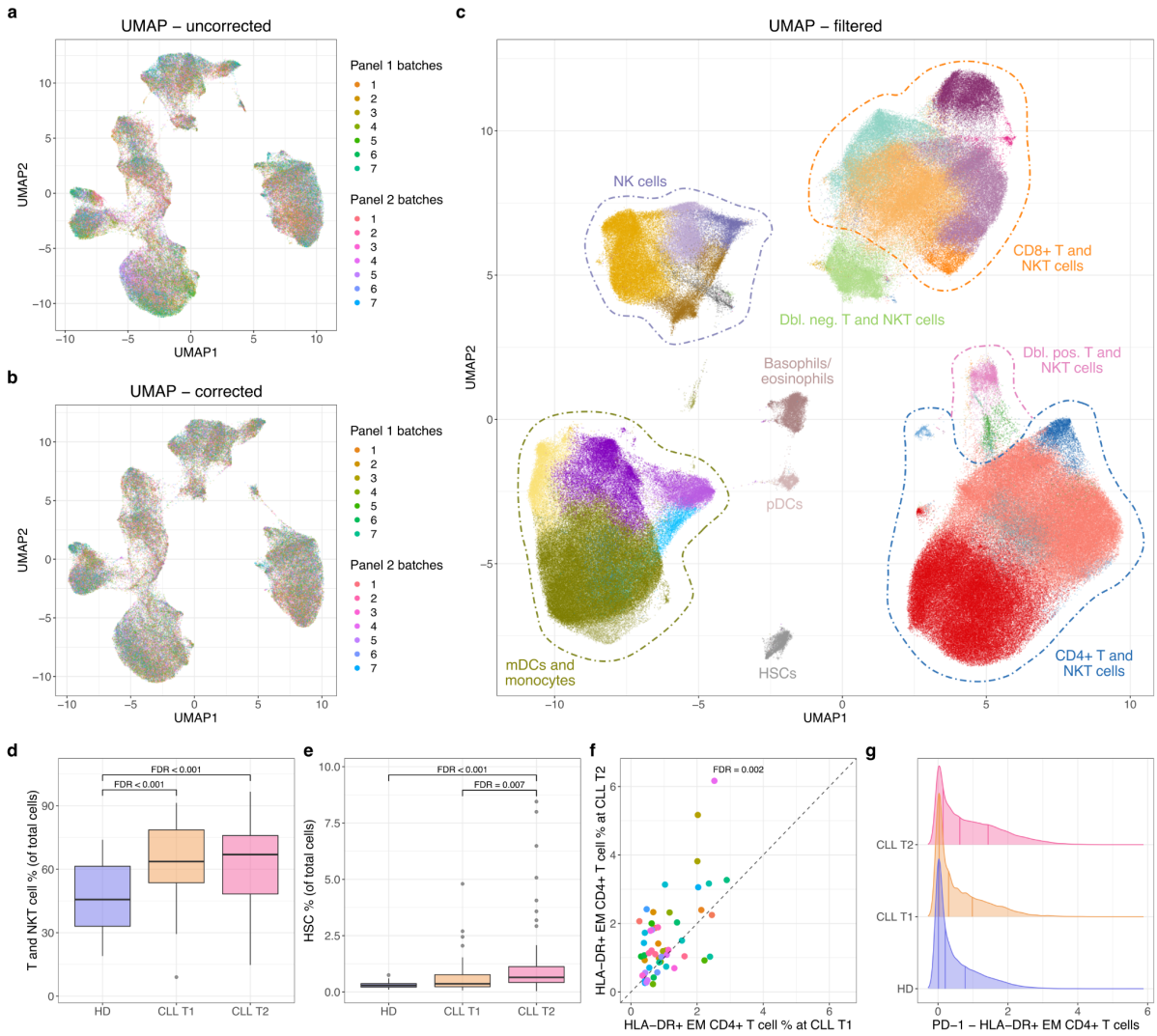

Figure 2. Integration and analysis of 128 CyTOF samples from 7 different batches and 2 different panels. **a**: UMAP-based on expression of the 12 overlapping lineage markers included in the final clustering for both panels 1 and 2 before any batch correction. Using ~100,000 cells with equal sampling from all batches. **b**: Same as in **a**, but after batch correction both within and between batches. **c**: UMAP for up to 2,000 cells from each of the 128 samples based on expression of the 23 clustering markers after removal of B, CLL, and poor-quality cells. Generated after panel

merging, clustering, and filtering, detailed labels in **Supplementary Figure 2**. **d-e**: Boxplots comparing the cell type proportion of two overall cell types between three sample groups: HD, CLL time point 1 (T1), and CLL time point 2 (T2). False discovery rates (FDR) for the differential abundance testing are added to the comparisons yielding significant (FDR < 0.01) results. Please note the use of different y axes. **f**: Scatter plot for the proportion of HLA-DR+ EM CD4+ T cells in paired CLL T1 and T2 samples. FDR value from differential abundance testing within the T and NKT cell compartment. **g**: Density plots for PD-1 expression levels in the HLA-DR+ EM CD4+ T cell population (panel 1 cells only) for the three sample groups: HD, CLL T1, and CLL T2.

cyCombine removes technical variance and maintains biological variance

Another scenario where batch correction is necessary is for the integration of external datasets. This is relevant when validating findings in public datasets or when performing meta-analysis of multiple existing datasets. To demonstrate cyCombine's capability to handle integration of data generated in different experimental setups, we integrated CyTOF samples from two different datasets. The two datasets were generated at different facilities, on different versions of the CyTOF instrument, with different panels of antibodies conjugated to different isotopes. Applying cyCombine reduced the EMD by 0.76, making the two datasets directly comparable, and with an MAD score of 0.04, indicating minimal loss of biological variance. As a testament to the robustness of cyCombine, the CLL samples being B cell depleted did not affect the batch correction, nor did the correction introduce B cells into the depleted batch (**Figure 3**).

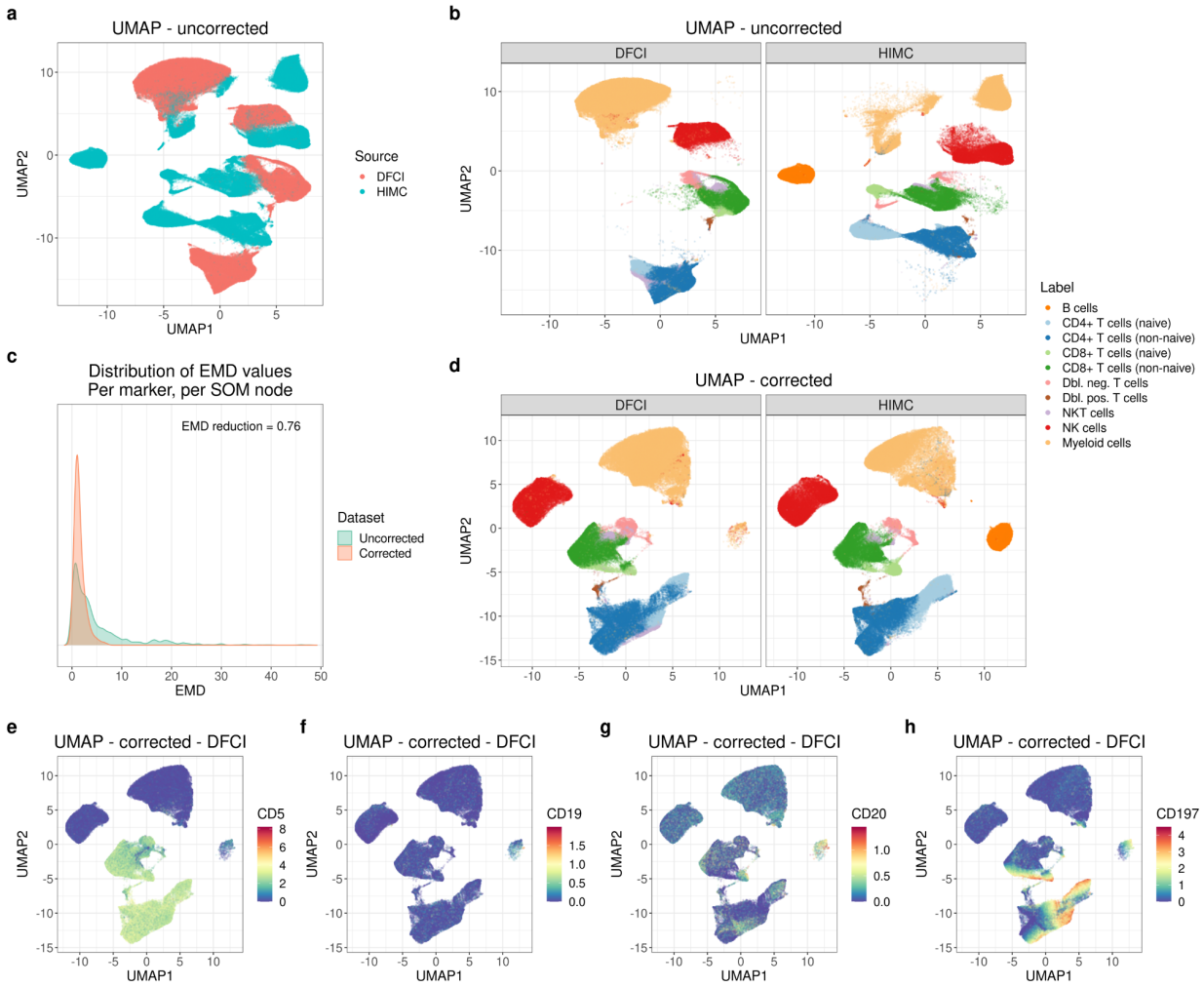

Figure 3. cyCombine rank-based batch correction for an HD sample from the HIMC dataset and an HD and a CLL sample from panel 1 of the DFCI data. **a**: UMAP for all cells from the two datasets based on expression of the 12 overlapping markers used for manual gating before batch correction. Colored by dataset. **b**: Same as in **a**, but faceted by dataset and colored by manually assigned labels. **c**: EMD density plots for uncorrected and corrected data. The EMD reduction was 0.76 and the MAD score was 0.04. **d**: UMAP for all cells from the two datasets based on expression of the 12 overlapping markers used for manual gating after batch correction. Colored by manually assigned labels (assigned before correction). **e-h**: Same as in **d**-DFCI, but colored by expression of CD5, CD19, CD20, and CD197 before batch correction.

When studying **Figure 3**, it is noticeable that a small cluster (0.5 %) appears in the DFCI set in the same UMAP position as the B cells from the HIMC set (11.9 %). We do not expect B cells in the DFCI set, so one could suspect that this means that B cells have been artificially introduced by cyCombine. However, when looking closer at these cells it becomes evident that their marker expression *before* correction is actually distinctly CLL cell-like, although with low CD19 expression explaining their presence after depletion. This fits with 82% of these cells originating from the CLL sample. While this observation makes biological sense, it highlights an important challenge when integrating cytometry: the breadth of the integrated dataset is limited by the overlapping markers

in the two panels. In this example, the CLL cells are mislabeled as myeloid due to lack of the CD5 marker for CLL cells and corresponding lack of typical myeloid markers such as CD11b.

cyCombine enables cross-platform data integration

As cyCombine is agnostic to marker distributions, it enables integration of datasets generated on entirely different platforms. This can be highly useful in cases where different single-cell technologies have been applied to assess the same samples and one wishes to directly integrate the results. It is also possible to integrate data from different studies, even when the data was generated using different technologies. To demonstrate this feature, we applied cyCombine to three healthy donor PBMC samples generated by CyTOF (HIMC dataset), CITE-seq (Illumina dataset), and spectral flow cytometry (Park et al. dataset[5]), respectively. While the raw data from the three data types assume distinct groupings in UMAP space (**Figure 4a**), batch correction using cyCombine makes the data directly comparable (**Figure 4b**). The resulting EMD reduction was 0.69 (**Figure 4c**) and the MAD score 0.07, and clustering and subpopulation labeling of cells qualitatively indicates that data are comparable (**Figure 4d**).

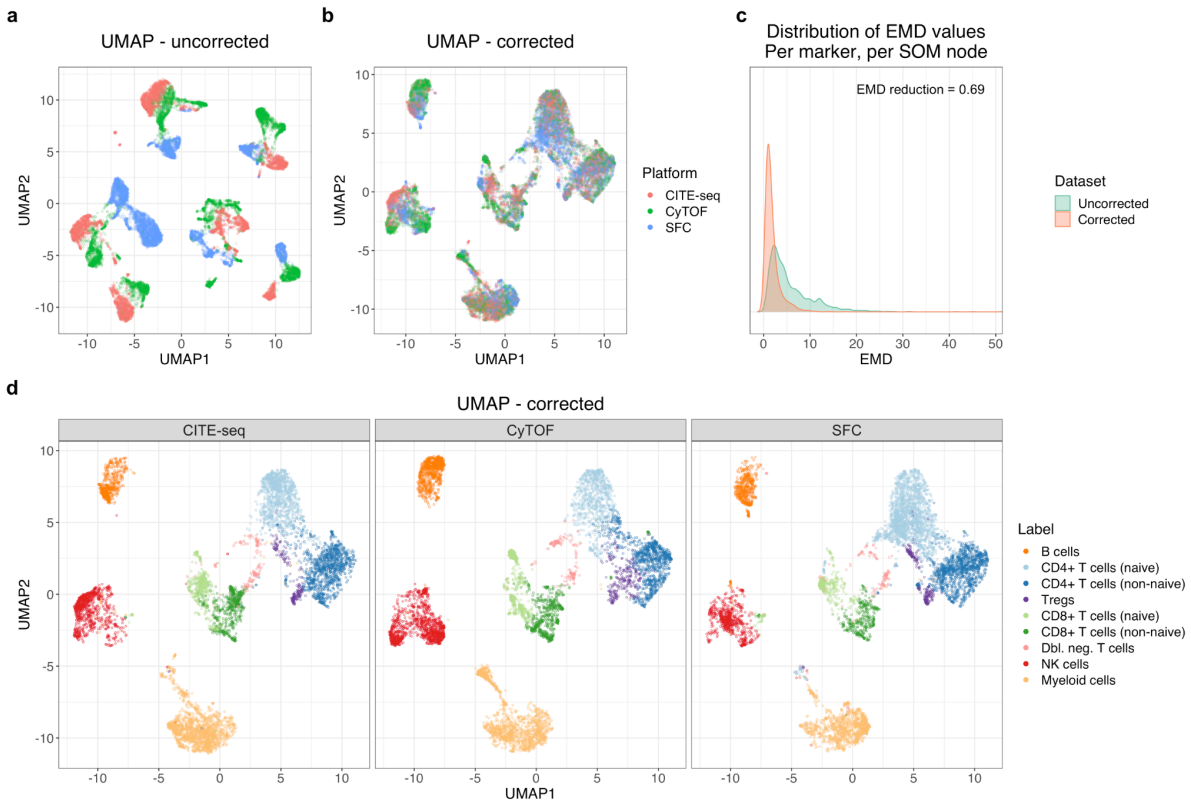

Figure 4. Cross-platform data integration. **a**: UMAP plot for uncorrected dataset consisting of 6,776 cells from each of the CITE-seq, CyTOF, and spectral flow cytometry (SFC) datasets. **b**: UMAP for the cells of **a** after batch correction with cyCombine. **c**: Density plots for EMDs calculated per marker, per SOM node for each of the pairwise comparisons between platforms. The SOM nodes used were those derived from corrected data. **d**: The corrected UMAP faceted by technology and colored by manually assigned labels determined on each dataset separately before correction.

cyCombine scales linearly with the number of cells

Another desirable application of cyCombine is for integration of very large cytometry datasets, e.g. from clinical trials or retrospective data from clinical diagnostics. Both the computation time and the memory requirements of cyCombine scale linearly with the number of cells and features, and, for example, the correction of 15 markers measured on 12,858,678 cells across 2 panels ran in 7 minutes on a standard laptop and required 10 GB of memory. This means that, while the memory requirements necessitate the use of a high performance computer, cyCombine can be applied to billions of cells in less than a day (for full runtime analysis see **Supplementary Figure 4** and **Supplementary Discussion**).

cyCombine outperforms all existing methods

Several tools for batch correction of both flow and mass cytometry data have been published. We tested the performance of all maintained, peer-reviewed tools: iMUBAC, CytoNorm, CytofRUV, and CytofBatchAdjust and compared their performance to cyCombine. To ensure a fair and broad comparison, we applied all tools to all the datasets used in the respective publications. As these tools have various limitations (e.g. designed to handle only one specific data type or condition, or designed to be dependent on technical replicates), each tool was tested only on datasets for which it was explicitly designed and tested by the authors. cyCombine was the only tool that could handle every single dataset in the test and showed superior performance for all of them when comparing the EMD reduction and MAD score (**Figure 6a** and **6b**).

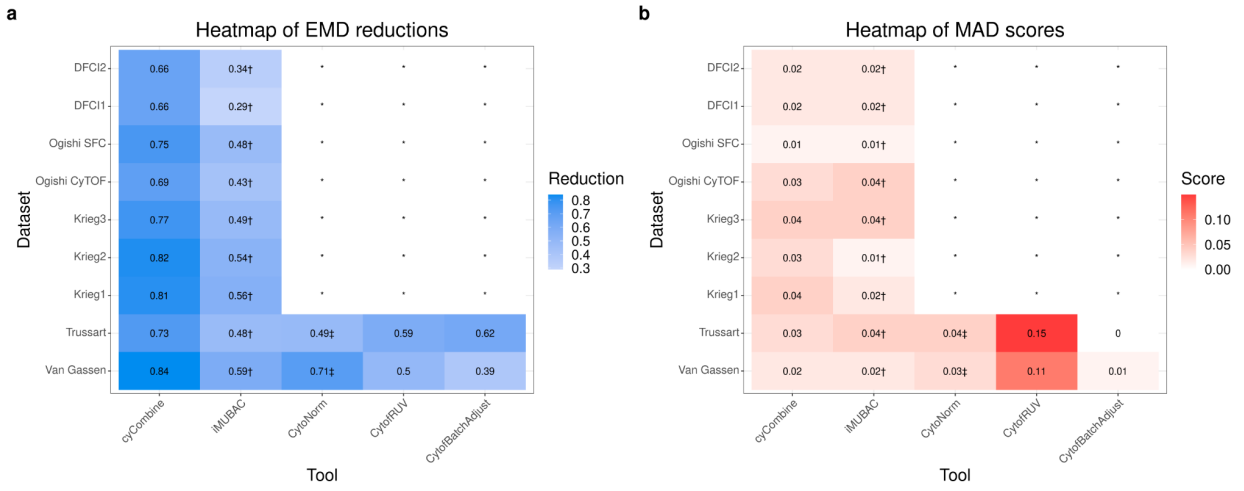

Figure 6. Performance evaluation of cyCombine and other previously published tools. **a**: Heatmap showing the EMD reductions of the batch correction tools run on various datasets. A reduction of 1 means a complete elimination of EMD, 0 means no change in EMD. The best-performing setting was selected for each tool. **b**: Heatmap showing the MAD scores of the batch correction tools run on various datasets. A score of 0 means a complete preservation of the biological variance of all markers in all batches. The best-performing setting was selected for each tool. In both **a** and **b**: * denotes that the tool is dependent on technical replicates, which is not available in the dataset. † denotes that the tool is only applied for healthy donor samples and utilizes subsampling. ‡ denotes

that the tool only corrects non-replicates and evaluations are performed on a subset of the full data.

Discussion

Deeper cytometric characterization of cell populations can have great implications, such as better diagnostics, development of novel therapeutics, and identification of important markers of immunity. However, a robust batch correction method is needed in order to fully realize the potential of single-cell cytometry. Correction of batch effects is often necessary to detect subtle biological variance in multi-batch experiments, and it is almost certainly a necessity for large-scale integration of data from different experiments.

In cyCombine, we handle cellular heterogeneity by applying careful overclustering of the data using a SOM. Co-clustering of data from all batches is enabled by an intermediary transformation of the expression values. The subsequent batch correction is performed using an empirical Bayes model, designed to reduce technical noise, while maintaining the biological signal. While others have previously used the EMD as a metric to measure the reduction in technical variance, we additionally describe the use of the MAD for quantifying the conservation of biological variance, which is a feature that has been overlooked in the majority of previously published methods.

Using these metrics, we demonstrate that cyCombine batch correction is quantifiably more accurate than existing tools, and through analysis of three different biologically relevant datasets, we highlight the high degree of flexibility and robustness of our method: cyCombine is independent of technical replicates across batches and makes no assumptions about homology of marker expression distributions. It is largely insensitive to sample and batch sizes, as it handles batch correction for as few as 8 cells in each SOM partition[11]. The SOM overclustering step ensures that both population abundances and cell phenotypes are retained, such that if batch effects are not present in a dataset, running the algorithm will not affect the expression values.

The primary limitations of cyCombine are inherited from ComBat, namely that batches and experimental conditions cannot be confounded. This means that at least one condition from each batch must be present in at least one other batch. Additionally, it is important to note that batch correction is only possible for features present in all samples.

Both the challenge and the possibilities presented here become no less relevant when both the rate of growth and heterogeneity of cytometry data increases as new technologies become more prevalent. cyCombine scales linearly with the number of cells, and we envision that cyCombine will catalyze an increase of large-scale analyses of cytometry data. Of particular interest are applications such as harmonization of clinical cytometry data, which may enable better application of machine learning algorithms for diagnostics, for example by enabling faster detection of minimal residual disease in hematological cancers. A range of use cases, including code and in-depth discussions, are available in the cyCombine vignettes: https://biosurf.org/cyCombine.

Data availability
CyTOF data will be made available on FlowRepository upon publication of this manuscript.

Code availability
The cyCombine R package is available on Github: https://github.com/biosurf/cyCombine/. Code to reproduce the analyses in this article is available at www.biosurf.org/cyCombine.

Author contributions
LRO and CBP conceived of the algorithm. CBP and SHD implemented the algorithm. CBP, LRO, MBB, MDL, and SHG designed the use cases. CBP, SHD, SHG, and LRO performed and interpreted the analyses. NP, LZR, TJK, JN, JAL, SHG, and CJW designed and generated the data for the chronic lymphocytic leukemia study. CBP, LRO, SHD, and SHG wrote the manuscript. All authors edited the manuscript.

Methods

The cyCombine package
cyCombine was designed with protein expression-based cytometry data in mind, and the functions for data preparation are made to handle FCS files. cyCombine assumes that the data has already been pre-gated (i.e. beads, dead cells, doublets, debris, etc. have been removed). When using the built-in functions, the data will be ArcSinh-transformed with a cofactor of choice (recommended cofactors are 5 for CyTOF, 150 for flow cytometry, and 6,000 for spectral flow cytometry). For CyTOF data, if counts are randomized, de-randomization is recommended[17]. However, the modules of cyCombine are not limited to data in FCS format, but are designed to work on any expression matrix that can be represented in an R data.frame - including CITE-seq protein expression data etc. cyCombine contains functions for importing FCS files, detection and correction of batch effects, plotting, evaluating batch correction, as well as performing panel merging. All functions are described in detail in the reference manual and the use case vignettes (https://biosurf.org/cyCombine).

The cyCombine batch correction module
cyCombine's batch correction module involves three separate steps: First, the expression of every marker is either Z-score normalized or converted to ranks, individually for each batch. Z-scoring is appropriate for similar datasets (e.g. multiple batches run on the same instrument with the same antibody clones and reporter molecules), whereas ranking tends to perform better for less similar datasets (e.g. data generated on different instruments, with different antibody-clones, different reporter molecules, or with different technologies). A self-organizing map (SOM)[12] is applied to the full normalized dataset. The grid size of the SOM should reflect the expected heterogeneity and result in a slight overclustering of the data. In cyCombine, the grid size defaults to 8x8, partitioning cells into 64 clusters. Then, the SOM node labels are assigned to the original expression value cells, and a per cluster batch correction is applied using ComBat[11]. The batch correction step can be performed with or without the use of a non-batch cofactor, e.g. phenotype or sample treatment. The cyCombine approach consequently allows for complex study designs, where not all conditions may be present in each batch, and where technical replicates were not

included. It is possible to perform batch correction in studies with more than two conditions, and one may integrate different datasets with only one overlapping condition while accounting for this imbalance. The only requirement is that at least one condition from each batch is present in at least one other batch.

Batch correction performance metrics

In order to evaluate the performance of the methods, we primarily applied an approach based on the EMD strongly inspired by Van Gassen et al. (2020)[18]. The EMD has previously been suggested to be a good metric for comparing protein expression distributions[18,19]. Briefly, the EMD was used to compare the distribution of each marker within SOM nodes across batches. Generally, the SOM nodes were determined post-batch correction using 8x8 grids, and the labels were transferred to the uncorrected data so each cell had the same label in both the uncorrected and corrected data. For an in-depth discussion, see the performance benchmarking vignette at https://biosurf.org/cyCombine. The distributions were binned with bin size = 0.1, and the EMDs for every marker for each pairwise batch comparison were computed. These scores were determined for both the uncorrected and corrected data, removing those values where both had an EMD < 2. The EMD reduction is given as:

$$EMD_{reduction} = \frac{\sum_{i=1}^{n}\left(EMD_{before_i} - EMD_{after_i}\right)}{\sum_{i=1}^{n} EMD_{before_i}}$$

Where *n* is the total number of comparisons (number of SOM nodes times the number of markers times the number of pairwise batch comparisons). Furthermore, we have developed a score that reflects the amount of variance removed during a batch correction process. The score is based on the median absolute deviation (MAD) and quantifies the variability of each marker in the dataset before and after correction. In practice, it is calculated very similarly to the EMD reduction: The MAD is calculated for the dataset after performing a SOM-based clustering, and is calculated per cluster, per marker, per batch. So, the MAD is calculated per batch, whereas the EMD calculations are performed for each pairwise batch-batch comparison. This means that the MAD score quantifies intra-batch effects of the correction, and the EMD reduction quantifies inter-batch effects. After calculating the MADs for both the corrected and uncorrected datasets, the MAD score is calculated as the median of the absolute difference in MAD per value:

$$MAD_{score} = median_{i=1}^{n}\left(\left|MAD_{before_i} - MAD_{after_i}\right|\right)$$

Where *n* is the total number of comparisons (number of SOM nodes times the number of markers times the number of batches).

The cyCombine panel merging module

cyCombine also contains two functions for marker imputation. One function is designed with panel merging in mind and imputes the expression values of non-overlapping markers across two datasets. It works by first doing a SOM-based (defaults to an 8x8 grid) clustering of the datasets

based on all of the overlapping markers. Then, for each cell in one of the datasets, the values for the missing markers are imputed by using the values from cells in the other dataset that fall within the same SOM node. The imputations are made by simulating a multi-dimensional kernel density estimate: Each cell's missing values are imputed by randomly drawing a cell from the other dataset and adding a Gaussian error, which is based on a draw from a Normal distribution with mean 0 and standard deviation corresponding to the bandwidth of each marker. However, if there are less than 50 cells from the other dataset within the SOM node, the values for the missing channels are set to NA as imputation would be unreliable.

The other function was made for salvaging a single channel within a dataset in selected batches. This can be useful in cases where one has a completely mis-stained marker in a single batch. It relies on the same principles, but instead of transferring information in one dataset to another, it utilizes intra-dataset batches.

Chronic lymphocytic leukemia cohort

Chronic lymphocytic leukemia (CLL) samples were obtained from the CLL Research Consortium (CRC) based at the University of California, San Diego, from patients who provided informed consent and as part of an institutional review board approved protocol. All samples were anonymized by the CRC. The dataset was generated at the Dana-Farber Cancer Institute (DFCI) and contained peripheral blood mononuclear cell (PBMC) samples from 20 healthy donors (5 from DFCI and 15 from HemaCare) and samples from 56 patients with CLL. The latter were sampled at two distinct time points (T1 and T2), the mean time between T1 and T2 was 58.7 months (sd = 47.4 months), and T2 was obtained close to first treatment (mean = 4.5 months, sd = 10.4 months) (**Figure 4**). For the 56 CLL patients, the mean age at diagnosis was 56.1 years (sd = 9.6 years), with healthy donors being age-matched (mean = 56.7 years, sd = 4.7 years).

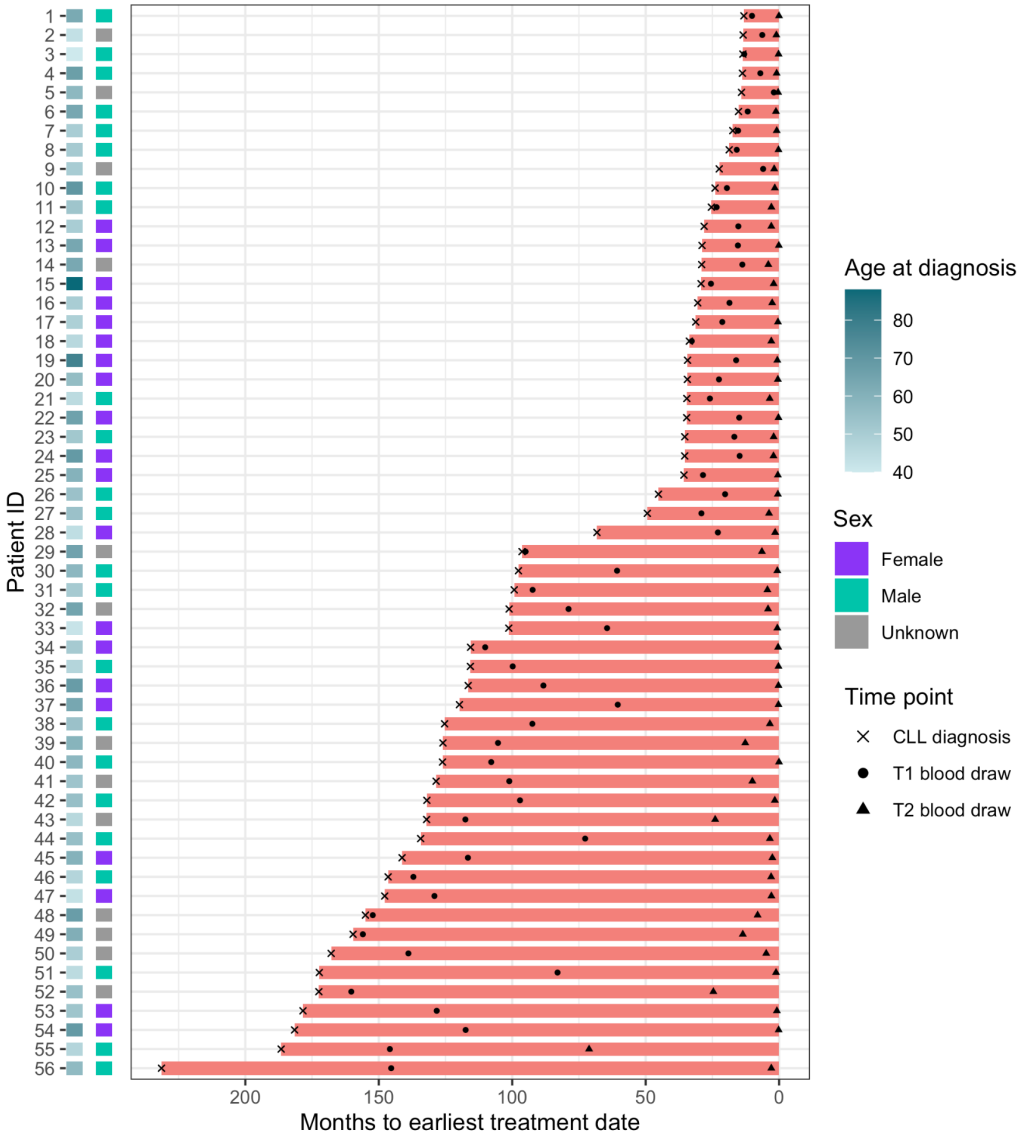

Figure 4. Time from CLL diagnosis to treatment initiation (months) for 56 CLL patients. Timing of blood draws for the T1 and T2 timepoints included in this study are indicated.

Immunophenotyping CLL cohort using mass cytometry

All patient and control PBMC samples were thawed in RPMI-1640 media (ThermoFisher) supplemented with 10% heat-inactivated FBS, sodium heparin (20 UI/mL) and 25 units/mL benzonase nuclease (Life Technologies and Sigma-Aldrich). Samples were subjected to B cell depletion using EasySep Human CD19 positive selection kit II (Stem Cell Technologies) before resuspension in RMPI and 10% FBS.

The samples were spun down and aspirated. 5 μM of cisplatin viability staining reagent (Fluidigm) was added for two minutes and then diluted with culture media. After centrifugation, Human TruStain FcX Fc receptor blocking reagent (BioLegend) was used at a 1:100 dilution final in cell

staining buffer (CSB) (PBS with 2.5 g/L bovine serum albumin and 100 mg/L of sodium azide, Sigma Aldrich) for 10 minutes followed by incubation with cell surface CyTOF antibody panels for 30 minutes (**Table 1**). All CyTOF antibodies were obtained from the Harvard Medical Area CyTOF Antibody Resource and Core (Lederer Lab, Brigham and Women's Hospital, Boston, MA).

16% stock paraformaldehyde (ThermoFisher Scientific) dissolved in PBS was used at a final concentration of 4% formaldehyde for 10 minutes in order to fix the samples before permeabilization with the FoxP3/Transcription Factor Staining Buffer Set (ThermoFisher Scientific). The samples were incubated with SCN-EDTA coupled palladium 20-sample barcoding reagents (Fluidigm) for 15 minutes, washed 3X in CSB, and then combined into a single 20 PBMC sample for subsequent staining. Conjugated intracellular CyTOF antibodies (**Table 1**) diluted in the permeabilization buffer from the FoxP3/Transcription Factor Staining Buffer Set were added into each tube and incubated for 30 minutes. Cells were then fixed with 1.6% formaldehyde for 10 minutes.

The samples were processed in seven batches per antibody panel, each batch containing both control and patient samples. During sample processing, some samples were excluded due to dead cells or having too few cells to apply both panels. The final dataset has measurements for a total of 128 samples, all of which were included in the staining with panel 1, and 112 that were also stained with panel 2. The 20 healthy donors were all stained with both panels. The CLL samples stained with panel 1 consisted of 52 samples at T1 and 56 (all patients) at T2. For panel 2, the numbers were 45 and 47, respectively. To identify single cell events, DNA was labeled for 20 minutes with an 18.75 μM iridium intercalator solution prior to acquisition. Samples were subsequently washed and reconstituted in cell acquisition solution (CAS) in the presence of EQ Four Element Calibration beads (Fluidigm) at a final concentration of $1x10^6$ cells/mL. Samples were acquired on a Helios CyTOF Mass Cytometer (Fluidigm).

Table 1. CyTOF panels for the CLL dataset. * = Batch 7: did not use - had none left. † = Batches 1-5: 1:200, batch 6: 1:400, batch 7: 1:200. Green background color denotes technical channels and yellow background indicates the 15 overlapping markers.

Channel	Panel 1 marker	Panel 1 clone	Panel 2 marker	Panel 2 clone
102Pd	BC1		BC1	
104Pd	BC2		BC2	
105Pd	BC3		BC3	
106Pd	BC4		BC4	
108Pd	BC5		BC5	
110Pd	BC6		BC6	
113In	CD20	2H7	CD20	2H7
115In	CD3	UCHT1	CD3	UCHT1
140Ce	EQ Beads		EQ Beads	

141Pr	CD27	O323		
142Nd	CD45RA	HI100	CD45RA	HI100
143Nd	CD279 (PD-1)	EH12.2H7	CD1c	L161
144Nd	CD5	UCHT2	CD5	UCHT2
145Nd	CD19	HIB19	CD19	HIB19
146Nd	CD14	M5E2	CD14	M5E2
147Sm	CD45RO	UCHL1	HLA-DR	L243
148Nd	Granzyme A	CB9		
149Sm	Granzyme K	GM26E7	CD1d	51.1
150Nd	FCRL6	7B7	CD11c	Bu15
151Eu	CD355 (CRTAM)	Cr24.1	CD123	6H6
152Sm	CD152 (CTLA4)	L3D10		
153Eu	CD69	FN50	JAK1	413104
154Sm	CD33	WM53	CD33	WM53
155Gd	CD4	RPA-T4	CD4	RPA-T4
156Gd	CD337 (NCR3)	P30-15		
157Gd			CD16	3G8
158Gd	CD8	RPA-T8	CD8	RPA-T8
159Tb	CD197 (CCR7)	G043H7	CD197 (CCR7)	G043H7
160Gd			IFNG	4S.B3
161Dy	LAG-3	874501	CD74	LN2
162Dy	CD56	NCAM16.2	CD56	NCAM16.2
163Dy	CD137 (4-1BB)	4B4-1	DR3 (TRAMP)	JD3
164Dy	CD161 (KLRB1)	HP-3G10	CD161 (KLRB1)	HP-3G10
165Ho	FoxP3	PCH101	FoxP3	PCH101
166Er	CD80	2D10	CD34	581
167Er	CD270 (HVEM)	122	IL23A	HLT2736
168Er	CD275 (ICOSL, B7-H2)*	136726	SMAD2	376520
169Tm	CD134 (OX40)	Ber-ACT35	CD11b	M1/70
170Er	CD278 (ICOS)	C398.4A		
171Yb	CD127	RDR5	CD184 (CXCR4)	12G5
172Yb	KLRG1	2F1/KLRG1	TGFBR2	16H2L4
173Yb	CD25	M-A251	FCER1A	AER-37
174Yb	HLA-DR	L243	TGFB1	TW4-2F8
175Lu	T-Bet	4B10	CD54 (ICAM1)	HA58
176Yb	XCL1†	Polyclonal	XCL1†	109001
191Ir	DNA1		DNA1	
193Ir	DNA2		DNA2	
195Pt	Viability		Viability	

209Bi			IL1RA	40007

Analysis of CLL cohort mass cytometry data

The raw FCS files were normalized to reduce signal deviation between samples over the course of multi-day batch acquisitions, utilizing the bead standard normalization method established by Finck et al.[20] as implemented in the premessa R package[21]. The normalized files were then compensated with a panel-specific spillover matrix to subtract cross-contaminating signals, utilizing the CyTOF-based compensation method established by Chevrier et al.[22] as implemented in CATALYST. These compensated files were then deconvoluted into individual sample files using a single-cell based debarcoding algorithm established by Zunder et al.[23] available in premessa. This was followed by pre-gating to live intact singlet cells using FlowJo version 10 (Tree Star Inc).

The pre-gated FCS files for each panel were read into R v. 4.0.0[24] using the cyCombine prepare_data function, using de-randomization and ArcSinh-transformation with cofactor = 5. The two panels consisted of a total of 6,027,290 and 6,831,388 cells. Subsequently, each panel was batch corrected using cyCombine with scaling and an 8x8 SOM grid using CLL/HD status as cofactor. After correction, all cells were clustered using an 8x8 SOM grid and the labels were transferred to the uncorrected data. The earth mover's distance (EMD) was calculated for each marker comparing the batches and the EMD reductions and MAD scores between corrected and uncorrected data were determined for each panel. The data from the two panels was then co-batch corrected using the 15 overlapping markers with scaling and an 8x8 SOM grid maintaining CLL/HD status as cofactor but using panel as batch. After co-correction, the 40 (19+21) non-overlapping markers were imputed using an 8x8 SOM grid and the resulting datasets were combined to a single 55-marker dataset.

The 55-marker data was then clustered using a 10x10 SOM grid[12] and ConsensusClusterPlus[13] using 23 markers: CD3, CD4, CD8, CD45RA, CD45RO, CD197, CD127, CD25, CD5, CD19, CD20, CD56, CD16, CD33, CD14, HLA-DR, CD123, CD1c, CD1d, CD11c, CD11b, FCER1A, and CD34. The result was extracted for 45 meta-clusters, and each of these was manually annotated based on its marker expression. Four of the clusters were labeled as either B cells (CD19+ CD20+) or CLL cells (CD19-lo CD20-lo CD5+), but because these populations can be considered cells that escaped the applied depletion, we removed those clusters from downstream analysis. Furthermore, four clusters displayed abnormal expression patterns, e.g. lack of lineage markers. When considering the mean viability stain for the clusters, it was observed that these four clusters all fell within the top-six highest values. This, together with the abnormal expression patterns, indicated that these clusters were composed of poor-quality cells, which we also excluded from further analysis. Finally, we iteratively clustered and merged the remaining 37 clusters based on high expression similarity as previously described[25], leaving a final set of 29 populations and 10,719,711 cells to study.

Differential abundance testing was carried out using an approach presented by Weber et al. (2019)[26] (testDA_voom). Each test included individual FDR-correction for the populations included, but no correction was performed between tests. Instead, a FDR-threshold of 0.01 was

used for significance. When relevant, the paired nature of the data was considered by using random effects. For differential expression testing *within* clusters, we analyzed the cell originating from each panel separately, meaning that no imputed values were included. The methodology for differential expression testing was also derived from the work by Weber et al. (2019)[26] (testDS_limma), in which medians serve as the foundation of the tests. Only markers not used for clustering were included in testing. Again, paired-ness was considered when appropriate, and an FDR-threshold of 0.01 was used.

HIMC healthy control sample

A single healthy donor PBMC sample (Human Immune Monitoring Center (HIMC) healthy donor, ctrls-001, MATLAB-normalized) was downloaded from FlowRepository (ID: FR-FCM-ZYAJ) and pre-gated to live intact singlets in FlowJo version 10 (Tree Star Inc). The 174,601 cells were processed in R using cyCombine with de-randomization and ArcSinh-transformation with a cofactor = 5. For the integration with the CLL dataset, this was followed by manual gating to 10 cell types based on the lineage markers, CD3, CD4, CD8, CD14, CD19, CD20, CD33, CD45RA, CD56, CD161, CD197, and HLA-DR. Unlabeled cells (n = 615) were discarded. For the three-datatype integration, the pre-gating was followed by clustering to 20 meta-clusters using a 6x6 SOM[12] grid and ConsensusClusterPlus[13] based on expression of 11 markers overlapping with the healthy donor spectral flow cytometry (SFC) and CITE-seq sets (CD3, CD4, CD8a, CD14, CD16, CD19, CD25, CD45RA, CD56, CD127, and PD-1). These clusters were annotated manually based on protein expression levels, and 8,932 cells were removed due to ambiguous expression patterns.

Flow cytometry dataset

The SFC dataset from Park et al.[5] was downloaded from FlowRepository (ID: FR-FCM-Z2QV). The dataset consists of samples from 4 healthy donor PBMCs, which were frozen and thawed, stained with 40 different antibodies in one panel, and analyzed using a 5-laser full spectrum flow cytometer (Cytek Biosciences Aurora).

Pre-processing was carried out in FlowJo version 10 (Tree Star Inc). The dataset was gated on lymphocytes, and singlets and non-debris were identified using forward and side-scatter. Dead cells were excluded using live/dead stains. Data from these gates were then exported in FCS format before further analysis in R: Using cyCombine, the data was loaded and transformed using ArcSinh with a cofactor = 6,000. A single sample (donor 303444) with 582,005 cells was selected and clustered to 20 meta-clusters using a 6x6 SOM[12] grid and ConsensusClusterPlus[13] based on expression of 11 markers overlapping with the healthy donor CyTOF and CITE-seq sets. The clusters were annotated manually based on protein expression levels, and 21,307 cells were removed due to ambiguous expression patterns.

Sequence barcoding-based dataset

The filtered feature/cell matrix from the “10k PBMCs from a Healthy Donor - Gene Expression and Cell Surface Protein” dataset was obtained from the 10X website (https://support.10xgenomics.com/single-cell-gene-expression/datasets/3.0.0/pbmc_10k_protein_v3). This data was generated on the PBMCs of a

single healthy donor stained with TotalSeq-B antibodies. It was sequenced on an Illumina NovaSeq and processed by Cell Ranger v. 3.0.0.

The TotalSeq expression matrix was processed in R using Seurat v. 4.0.0[27]. First, cells were filtered to maintain only those expressing between 200 and 2800 genes, having less than 10,000 detected RNA molecules and 20,000 detected protein molecules, and with a mitochondrial gene percentage below 10, leaving 6,949 cells for analysis. The protein portion of the data was normalized, scaled, and dimensionality reduced to the 11 markers overlapping with the CyTOF and SFC datasets, before applying Louvain clustering at a resolution of 0.2. The 12 resulting clusters were manually annotated based on the expression levels of the 11 clustering proteins. Two clusters were considered to be doublets and excluded from the downstream integration, leaving 6,776 cells.

Integration of CLL and HIMC healthy donor sample

For the integration with the HIMC healthy donor sample, two samples from the DFCI set (one CLL and one HD) from panel 1, batch 5 were selected (before any batch correction was applied) and manually gated to 10 cell types based on 12 lineage markers: CD3, CD4, CD8, CD14, CD19, CD20, CD33, CD45RA, CD56, CD161, CD197, and HLA-DR. Unlabeled cells (n = 4,353) were considered to be representative of the low-quality cells, and were discarded along with any cells labeled as B cells, since these were residual cells resulting from incomplete depletion. The HIMC sample was likewise gated to 10 populations using the same 12 lineage markers. This resulted in a total of 352,210 cells, with 17 overlapping markers between the datasets (CD3, CD4, CD8, CD14, CD19, CD20, CD25, CD27, CD33, CD45RA, CD56, CD127, CD161, CD197, HLA-DR, ICOS, and PD-1). Datasets were batch corrected using cyCombine with an 8x8 SOM grid with the rank normalization method (and average ties method). Each set was considered a batch, and the HD/CLL status was used as a cofactor. The result of the batch correction was evaluated with the EMD reduction and MAD score as well as visual inspection of UMAP plots comparing the location of each cell type (which was assigned separately) across datasets.

Integration of cross-platform datasets

The HIMC CyTOF sample, the SFC sample, and the CITE-seq data were batch corrected together following the pre-processing described in the section for each set. Before batch correction, each set was downsampled to 6,776 cells and to the 11 overlapping protein markers. This was followed by cyCombine batch correction with an 8x8 SOM grid with the rank normalization method (and average ties method). Each dataset was considered a batch and no cofactors were considered. The result of the batch correction was evaluated with the EMD reduction and MAD score as well as UMAP plots comparing the location of each cell type (which was assigned separately) across datasets.

Benchmarking

We compared the performance of the cyCombine batch correction module with four batch correction algorithms designed to work with mass cytometry data: CytoNorm[18], CytofRUV[28], iMUBAC[29], and CytofBatchAdjust[30]. Other tools exist, both developed for flow and mass cytometry, including gaussNorm and fdaNorm[31,32], which the authors state are no longer

supported, and the tools cydar[33], BatchEffectRemoval[34], BatchEffectRemoval2018[35], SAUCIE[36], and swiftReg[37], which are not included due to either not being peer-reviewed, not being maintained, requiring a license, or being designed to work only on very specific cases, such as harmonizing two technical replicates. We tested each included tool on the datasets from the original publications and the set of datasets from other publications deemed to be suitable by the authors of each tool; i.e. some tools require technical replicates and not all datasets include these. Furthermore, we only test each tool on datasets from platforms for which the use is demonstrated in the original publication. For tools with multiple tested settings, the setting with the best overall performance based on both the EMD reduction and MAD score was recorded.

All five included tools were run on the CyTOF datasets originally presented in the CytoNorm and CytofRUV papers. We will refer to these two sets as the Van Gassen and the Trussart data, respectively. Additionally, we batch corrected six CyTOF datasets and one SFC set without technical replicates using cyCombine and iMUBAC. These datasets are the DFCI panel 1 and panel 2 sets, and five datasets presented in the iMUBAC article: Each of the three panels of the Krieg dataset, as well as a CyTOF and a SFC set originally generated for iMUBAC, which we refer to as $Ogishi_{CyTOF}$ and $Ogishi_{SFC}$. An overview is presented in **Table 2**. All CyTOF datasets were ArcSinh-transformed with a cofactor = 5 for processing with all tools.

Table 2. Datasets used for benchmarking study. * = Counting replicates as distinct samples.

Dataset	Instrument	Samples	Batches	Conditions	Cells (million)	Markers	FlowRepository ID	Originally used for tool
Van Gassen[38]	CyTOF2.0	40*	10	2	6.2	37	FR-FCM-Z247	CytoNorm[18]
Trussart[28]	Helios	24*	2	2	8.6	31	FR-FCM-Z2L2	CytofRUV[28]
Krieg1[39]	Helios(2.1)	60	4	3	1.1	30	FR-FCM-ZY34	iMUBAC[29]
Krieg2[39]	Helios(2.1)	60	4	3	1.7	26	FR-FCM-ZY34	iMUBAC
Krieg3[39]	Helios(2.1)	60	4	3	0.3	25	FR-FCM-ZY34	iMUBAC
$Ogishi_{CyTOF}$[29]	Helios	57*	7	3	12.4	38	FR-FCM-Z3YK	iMUBAC
$Ogishi_{SFC}$[29]	Aurora	14	2	3	9.7	18	FR-FCM-Z3YL	iMUBAC
DFCI1	Helios	128	7	2	6.0	36	TBA	cyCombine
DFCI2	Helios	112	7	2	6.8	34	TBA	cyCombine

The Van Gassen dataset[18] consists of 40 samples from two healthy controls. They comprise unstimulated and stimulated samples each run 10 times (10 batches). 37 protein markers were measured. The Trussart dataset[28] consists of 24 samples from nine healthy controls (HC) and three CLL patients, each run twice (two batches). 31 protein markers were measured. The FCS files were pre-processed with bead normalization and debarcoding according to the script from the CytofRUV supplementary files (using CATALYST). The Krieg1, Krieg2, and Krieg3 datasets[39] comprise 30, 26, and 25 markers, and each contain 60 samples. They were, according to the

original publication, processed as four experimental batches. Three conditions are considered: Healthy donors (*n* = 20), responders (*n* = 22), and non-responders (*n* = 18) to anti-PD-1 immunotherapy. Each condition is included in each of the four batches. The dataset was pre-processed according to the instructions in the iMUBAC article: DNA and viability intercalators were used to exclude dead cells, doublets, and debris with the prepSCE function from iMUBAC. The $\text{Ogishi}_{\text{CyTOF}}$ dataset[29] contains measurements on 38 protein markers and consists of 57 samples in seven batches. A total of three conditions were included: Healthy (*n* = 50), MSMD (*n* = 5), and Salmonellosis (*n* = 2). Some of the healthy samples are biological replicates. The dataset was pre-processed according to the instructions in the iMUBAC article: DNA and viability intercalators were used to exclude dead cells, doublets, and debris. The $\text{Ogishi}_{\text{SFC}}$ dataset[29] measured 18 protein markers across 14 samples in two batches. A total of three conditions were included: Healthy donors (*n* = 11) and two types of autoimmune disease (*n* = 1 and *n* = 2). The dataset was pre-processed according to the instructions in the iMUBAC article: The viability stain was used to exclude dead cells and logicle transformation was used.

When running CytoNorm, we used FlowSOM clustering with a 10x10 grid and 25 final clusters (no downsampling). The batch effects were modelled using 101 quantiles. All protein markers were included. For the Van Gassen set, the 20 samples from healthy control 1 were used to model batch effects and the 20 samples from healthy control 2 were normalized. Evaluation of batch effect reduction was carried out using only the samples from healthy control 2. For the Trussart dataset, the CLL2 and HC1 samples were used as the technical replicates (training data). The remaining 20 samples were used as validation data and the evaluation of batch effect reduction was carried out using only the HC2-9, CLL1, and CLL3 samples. Corrected values were capped at 300 to avoid problems with very large values during evaluation.

For running CytofRUV, we used clustering with 20 clusters on lineage markers only (24 for Van Gassen and 19 for Trussart). All markers were corrected at varying values of $k = \{5,10,15,20\}$. For the Van Gassen set, all healthy control 1 samples were used as technical replicates (two sets of 10 samples each). For the Trussart set, the CLL2 and HC1 samples were used as the technical replicates. All samples were included in the evaluation.

For running CytofBatchAdjust, all files were renamed according to the tool requirements. For Van Gassen, PTLG021 was used as the reference batch and the unstimulated healthy control 1 samples were used as anchors. We tested CytofBatchAdjust with *method* = {95p, SD, quantile}. For the Trussart set, HC1 was used as the anchor sample and RUV1b samples as reference batch. All markers were used for correction and all samples were used in evaluation. Corrected values were capped at 300 to avoid problems with very large values during evaluation.

iMUBAC was run largely according to the details in the original article. For all datasets, only healthy donors were included in correction, and downsampling to 200,000 cells for each batch was applied for all datasets, except for the Krieg3 dataset, for which we downsampled to 50,000 cells per batch, and the $\text{Ogishi}_{\text{SFC}}$ set, for which 500,000 cells per batch were included. For the $\text{Ogishi}_{\text{CyTOF}}$ set, only 47 local healthy donor samples were included as in the original publication

(travel/family controls excluded). All evaluations were based solely on the downsampled datasets using all markers.

cyCombine was generally run on all available samples using the conditions stated in the presentation of each dataset. We ran cyCombine with *norm_method* = {scale, rank} on the full datasets with all markers.

Runtime and memory requirements

Several of the evaluated tools ran directly on FCS files; therefore, running these tools on a range of different sizes required storing downsampled versions of the original FCS files in new ones. This was done by loading the original FCS files, disregarding non-overlapping columns, sampling to the predefined sample-sizes, and storing the resulting data in respective folders. By storing the data this way, it was ensured that all tools were run on the same data at each data size. The runtime and memory usage were measured for each tool for every sample size using the UNIX command “time -v”. The “Maximum resident set size” and the “elapsed” parameters in the output defined the memory usage and runtime, respectively. The test was performed on 40 cores (although none of the tools are fully parallelized, some sub functions are) on an HPE Apollo 2000 system with up to 192 GB PC4 2933 RAM. The “standard laptop” was a 2018 MacBook Pro with 16 GB 2400 MHz DDR4 memory and a 2.6 GHz 6-Core Intel Core i7 processor.

Plots

UMAPs were generated using uwot v. 0.1.9[40]. Plots were generated using ggridges v. 0.5.2[41] and ggplot2 v. 3.3.3[42], and patchwork v. 1.1.1[43] was used for combining plots.

References

1. Jaye, D. L., Bray, R. A., Gebel, H. M., Harris, W. A. C. & Waller, E. K. Translational applications of flow cytometry in clinical practice. *J. Immunol.* **188**, 4715–4719 (2012).
2. Hartmann, F. J. *et al.* Single-cell metabolic profiling of human cytotoxic T cells. *Nat. Biotechnol.* **39**, 186–197 (2021).
3. Bandura, D. R. *et al.* Mass cytometry: technique for real time single cell multitarget immunoassay based on inductively coupled plasma time-of-flight mass spectrometry. *Anal. Chem.* **81**, 6813–6822 (2009).
4. Spitzer, M. H. & Nolan, G. P. Mass cytometry: single cells, many features. *Cell* **165**, 780–791 (2016).
5. Park, L. M., Lannigan, J. & Jaimes, M. C. OMIP-069: Forty-Color Full Spectrum Flow Cytometry Panel for Deep Immunophenotyping of Major Cell Subsets in Human Peripheral Blood. *Cytometry A* **97**, 1044–1051 (2020).
6. Stoeckius, M. *et al.* Simultaneous epitope and transcriptome measurement in single cells. *Nat. Methods* **14**, 865–868 (2017).

7. Schoof, E. M. *et al.* A Quantitative Single-Cell Proteomics Approach to Characterize an Acute Myeloid Leukemia Hierarchy. *BioRxiv* (2019) doi:10.1101/745679.
8. Budnik, B., Levy, E., Harmange, G. & Slavov, N. SCoPE-MS: mass spectrometry of single mammalian cells quantifies proteome heterogeneity during cell differentiation. *Genome Biol.* **19**, 161 (2018).
9. Brunner, A.-D. *et al.* Ultra-high sensitivity mass spectrometry quantifies single-cell proteome changes upon perturbation. *BioRxiv* (2020) doi:10.1101/2020.12.22.423933.
10. Rybakowska, P., Alarcón-Riquelme, M. E. & Marañón, C. Key steps and methods in the experimental design and data analysis of highly multi-parametric flow and mass cytometry. *Comput. Struct. Biotechnol. J.* **18**, 874–886 (2020).
11. Johnson, W. E., Li, C. & Rabinovic, A. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics* **8**, 118–127 (2007).
12. Kohonen, T. Self-organized formation of topologically correct feature maps. *Biol. Cybern.* **43**, 59–69 (1982).
13. Wilkerson, M. D. & Hayes, D. N. ConsensusClusterPlus: a class discovery tool with confidence assessments and item tracking. *Bioinformatics* **26**, 1572–1573 (2010).
14. Nunes, C. *et al.* Expansion of a CD8(+)PD-1(+) replicative senescence phenotype in early stage CLL patients is associated with inverted CD4:CD8 ratios and disease progression. *Clin. Cancer Res.* **18**, 678–687 (2012).
15. Elston, L. *et al.* Increased frequency of CD4+ PD-1+ HLA-DR+ T cells is associated with disease progression in CLL. *Br. J. Haematol.* **188**, 872–880 (2020).
16. Riches, J. C. *et al.* T cells from CLL patients exhibit features of T-cell exhaustion but retain capacity for cytokine production. *Blood* **121**, 1612–1621 (2013).
17. Olsen, L. R., Leipold, M. D., Pedersen, C. B. & Maecker, H. T. The anatomy of single cell mass cytometry data. *Cytometry A* **95**, 156–172 (2019).
18. Van Gassen, S., Gaudilliere, B., Angst, M. S., Saeys, Y. & Aghaeepour, N. Cytonorm: A normalization algorithm for cytometry data. *Cytometry A* **97**, 268–278 (2020).
19. Orlova, D. Y. *et al.* Earth mover's distance (EMD): A true metric for comparing biomarker expression levels in cell populations. *PLoS ONE* **11**, e0151859 (2016).
20. Finck, R. *et al.* Normalization of mass cytometry data with bead standards. *Cytometry A* **83**, 483–494 (2013).
21. Gherardini, P. F. *premessa: R package for pre-processing of flow and mass cytometry data*. (R package version 0.2.6, 2021).
22. Chevrier, S. *et al.* Compensation of signal spillover in suspension and imaging mass cytometry. *Cell Syst.* **6**, 612-620.e5 (2018).
23. Zunder, E. R. *et al.* Palladium-based mass tag cell barcoding with a doublet-filtering scheme and single-cell deconvolution algorithm. *Nat. Protoc.* **10**, 316–333 (2015).

24. R Core Team. R: A Language and Environment for Statistical Computing. (2021).
25. Pedersen, C. B. & Olsen, L. R. Algorithmic Clustering Of Single-Cell Cytometry Data-How Unsupervised Are These Analyses Really? *Cytometry A* **97**, 219–221 (2020).
26. Weber, L. M., Nowicka, M., Soneson, C. & Robinson, M. D. diffcyt: Differential discovery in high-dimensional cytometry via high-resolution clustering. *Commun. Biol.* **2**, 183 (2019).
27. Stuart, T. *et al.* Comprehensive Integration of Single-Cell Data. *Cell* **177**, 1888-1902.e21 (2019).
28. Trussart, M. *et al.* Removing unwanted variation with CytofRUV to integrate multiple CyTOF datasets. *elife* **9**, (2020).
29. Ogishi, M. *et al.* Multibatch cytometry data integration for optimal immunophenotyping. *J. Immunol.* **206**, 206–213 (2021).
30. Schuyler, R. P. *et al.* Minimizing batch effects in mass cytometry data. *Front. Immunol.* **10**, 2367 (2019).
31. Hahne, F. *et al.* Per-channel basis normalization methods for flow cytometry data. *Cytometry A* **77**, 121–131 (2010).
32. Finak, G. *et al.* High-throughput flow cytometry data normalization for clinical trials. *Cytometry A* **85**, 277–286 (2014).
33. Lun, A. T. L., Richard, A. C. & Marioni, J. C. Testing for differential abundance in mass cytometry data. *Nat. Methods* **14**, 707–709 (2017).
34. Shaham, U. *et al.* Removal of batch effects using distribution-matching residual networks. *Bioinformatics* **33**, 2539–2546 (2017).
35. Shaham, U. Batch Effect Removal via Batch-Free Encoding. *BioRxiv* (2018) doi:10.1101/380816.
36. Amodio, M. *et al.* Exploring single-cell data with deep multitasking neural networks. *Nat. Methods* **16**, 1139–1145 (2019).
37. Rebhahn, J. A., Quataert, S. A., Sharma, G. & Mosmann, T. R. SwiftReg cluster registration automatically reduces flow cytometry data variability including batch effects. *Commun. Biol.* **3**, 218 (2020).
38. Aghaeepour, N. *et al.* An immune clock of human pregnancy. *Sci. Immunol.* **2**, (2017).
39. Krieg, C. *et al.* High-dimensional single-cell analysis predicts response to anti-PD-1 immunotherapy. *Nat. Med.* **24**, 144–153 (2018).
40. Melville, J. *uwot: The Uniform Manifold Approximation and Projection (UMAP) Method for Dimensionality Reduction*. (R package version 0.1.9, 2020).
41. Wilke, C. O. *ggridges: Ridgeline Plots in "ggplot2."* (R package version 0.5.2, 2020).
42. Wickham, H. *ggplot2: Elegant Graphics For Data Analysis*. (Springer-Verlag New York, 2016).

43. Pedersen, T. L. *patchwork: The Composer of Plots*. (R package version 1.1.1, 2020).