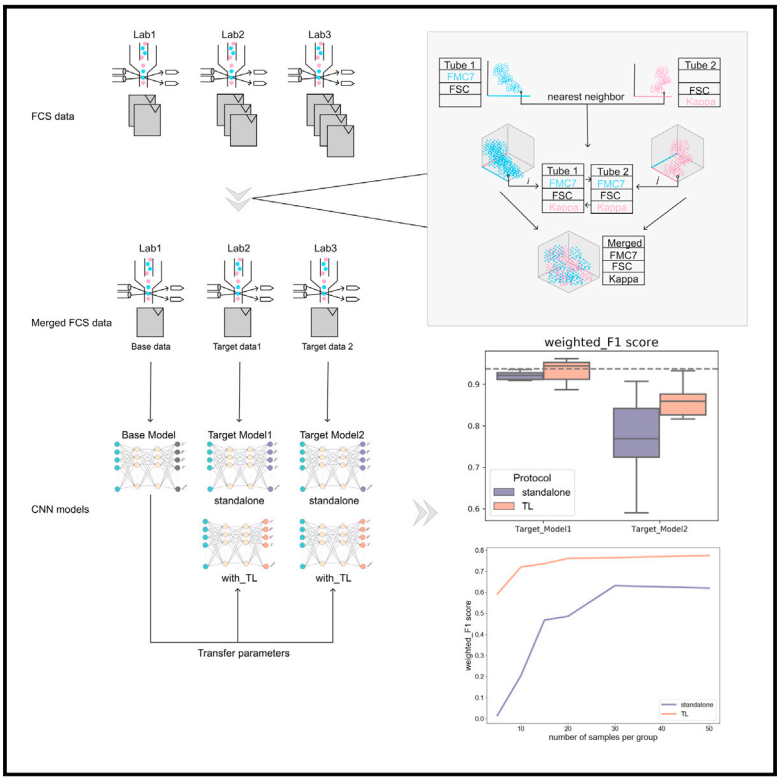
Article

Patterns

Knowledge transfer to enhance the performance of deep learning models for automated classification of B cell neoplasms

Graphical abstract



Highlights

- Device capabilities and diagnostic approaches differ greatly in lymphoma MFC panels
- Single laboratories generate too little data to train an AI model with high accuracy
- Transfer learning across panels increases classification performance significantly
- Merging MFC data from multiple tubes per sample increases the model's transferability

Authors

Nanditha Mallesh, Max Zhao, Lisa Meintker, ..., Peter Brossart, Stefan W. Krause, Peter M. Krawitz

Correspondence

pkrawitz@uni-bonn.de

In brief

In lymphoma diagnostics, artificial intelligence (AI) can save time and cost by improving the accuracy in disease subtyping with multi-parameter flow cytometry (MFC) data. So far, AI has been limited to the MFC protocol that was used to train the models. We present a framework to extend AI to multiple MFC protocols using transfer learning (TL). We demonstrate that TL in combination with MFC data merging achieves higher performance for smaller training sizes.

Mallesh et al., 2021, Patterns 2, 100351
October 8, 2021
https://doi.org/10.1016/j.patter.2021.100351

CellPress

Patterns

CellPress
OPEN ACCESS

Article

Knowledge transfer to enhance the performance of deep learning models for automated classification of B cell neoplasms

Nanditha Mallesh,[1,9] Max Zhao,[1,2,9] Lisa Meintker,[8] Alexander Höllein,[3,4] Franz Elsner,[5] Hannes Lüling,[5] Torsten Haferlach,[3] Wolfgang Kern,[3] Jörg Westermann,[6] Peter Brossart,[7] Stefan W. Krause,[8] and Peter M. Krawitz[1,10,*]

[1]Institute for Genomic Statistics and Bioinformatics, University Bonn, Bonn, Germany
[2]Institute of Human Genetics and Medical Genetics, Charité University Hospital, Berlin, Germany
[3]MLL Munich Leukemia Laboratory, Munich, Germany
[4]Red Cross Hospital Munich, Munich, Germany
[5]res mechanica GmbH, Munich, Germany
[6]Department of Hematology, Oncology and Tumor Immunology, Charité-Campus Virchow Clinic and Labor Berlin Charité Vivantes, Berlin, Germany
[7]Department of Oncology, Hematology, Immuno-oncology and Rheumatology, University Hospital of Bonn, Bonn, Germany
[8]Department of Medicine 5, Universitätsklinikum Erlangen, Erlangen, Germany
[9]These authors contributed equally
[10]Lead contact
*Correspondence: pkrawitz@uni-bonn.de
https://doi.org/10.1016/j.patter.2021.100351

THE BIGGER PICTURE Multi-parameter flow cytometry (MFC) is a critical tool in leukemia and lymphoma diagnostics. Advances in cytometry technology and diagnostic standardization efforts have led to an ever-increasing volume of data, presenting an opportunity to use artificial intelligence (AI) in diagnostics. However, the MFC protocol is prone to changes depending on the diagnostic workflow and the available cytometer. The changes to the MFC protocol limit the deployment of AI in routine diagnostics settings. We present a workflow that allows existing AI to adapt to multiple MFC protocols. We combine transfer learning (TL) with MFC data merging to increase the robustness of AI. Our results show that TL improves the performance of AI and allows models to achieve higher performance with less training data. This gain in performance for smaller training data allows for an already deployed AI to adapt to changes without the need for retraining a new model that requires more training data.

Development/Pre-production: Data science output has been rolled out/validated across multiple domains/problems

SUMMARY

Multi-parameter flow cytometry (MFC) is a cornerstone in clinical decision making for leukemia and lymphoma. MFC data analysis requires manual gating of cell populations, which is time-consuming, subjective, and often limited to a two-dimensional space. In recent years, deep learning models have been successfully used to analyze data in high-dimensional space and are highly accurate. However, AI models used for disease classification with MFC data are limited to the panel they were trained on. Thus, a key challenge in deploying AI into routine diagnostics is the robustness and adaptability of such models. This study demonstrates how transfer learning can be applied to boost the performance of models with smaller datasets acquired with different MFC panels. We trained models for four additional datasets by transferring the features learned from our base model. Our workflow increased the model's overall performance and, more prominently, improved the learning rate for small training sizes.

INTRODUCTION

Multi-parameter flow cytometry (MFC) is a powerful and high-throughput technique that allows for rapid quantification of markers on cells in suspension.[1] Today, it is a critical step in both research and clinical decision making for leukemia[2,3] and other hematological diseases. The increasing number of parameters that can be measured with modern devices, a widely adopted flow cytometry standard (FCS) for data by all manufacturers, and the possibility of data anonymization makes MFC ideal for deep learning on shared data. Despite all this, most data are still analyzed manually, including gating cell populations of interest in a two-dimensional scatterplot, which is time-consuming and subjective.[4,5] In recent years, more advanced computational methods involving deep learning have become available that can accurately classify disease subtypes based on cell type identification from cytological images[6] and perform automated classification of MFC data into diagnosis labels.[7,8] However, such models used for MFC analysis are limited to the MFC panel they are trained on and do not produce the same performance on a different MFC panel.

The flow cytometry panel design across various laboratories varies depending on the markers to be analyzed and the cytometer available. In many cases, the number of markers needed to be analyzed exceeds the number that the cytometer can measure in a single run. Standard practice is to aliquot a sample into multiple tubes, each of which often includes a set of shared or backbone markers.[9] This process is standard for modern clinical diagnostic of MFC data, especially when immunophenotyping leukemia and lymphoma. Furthermore, the choice of markers depends on the diagnostic workflow and is not standardized. These differences result in different antibody (MFC) panels being used in different laboratories. Thus, artificial intelligence (AI) models must be robust and adapt to different MFC panels across laboratories and within the same laboratory when the diagnostic panel is modified due to transition to a new cytometer.

This study extends our previous model,[8] where we trained an AI model to classify seven B cell neoplasm subtypes plus healthy controls for a nine-color panel to work with multiple MFC panels by employing transfer learning (TL). TL is a technique to improve the performance of a new task by transferring knowledge from a related task that has already been learned.[10] The new task (target task) to be learned usually has a smaller dataset than the base data with which the related task (base task) was learned. However, the target and base data do not generally change in composition. In our case, while both the base and target tasks to be learned are the same (classifying B cell neoplasms into diagnosis labels), the MFC protocol with which a sample is acquired is subject to inter-laboratory variability and changes over time in terms of the number of tubes per sample, markers measured, marker-fluorochrome conjugates, and other protocol parameters.

To handle the differences across MFC panels and achieve maximum knowledge transfer, we merge FCS data from individual tubes of a sample into a single combined FCS file using the nearest neighbor (NN) method. This method assumes that a cell in one tube is identical to its NN in another tube in terms of the shared markers and can thus be used to impute missing marker values.[11,12] The expression vectors of all the NNs across tubes are merged, creating a single, high-dimension matrix of cellular expression across all tubes. NN merging has proven effective as part of classification pipelines,[9,13] while other merging methods are better suited for deep profiling.[14] We use NN merge in conjunction with TL to generalize our model and achieve a higher learning rate with fewer training samples.

RESULTS

Overview

An overview of the TL process is shown in Figure 1. Before knowledge transfer, we merge multiple aliquots (tubes) per sample into a single FCS data file using the NN merge approach. In our previously published model,[8] we processed individual tubes of each sample separately, resulting in a convolution neural network (CNN) architecture that depends on the number of tubes per sample. Without the initial merge step, such a network's transferability between datasets with a different number of tubes per sample is very low—we can only transfer knowledge from the dense layers (Figure S1). Merging multiple aliquots allows for maximum transfer between the networks—weights from all layers can now be transferred. Next, a self-organizing map (SOM) is generated for each merged sample. The generated SOM node weights, which are n-dimensional vectors of the original FCS data arranged on a two-dimensional grid, are used as input to the CNN that generates class predictions. We use SOMs to process FCS data and convert it into a suitable input for CNN. While SOMs are appropriate for our workflow, there are alternatives to SOM, such as UMAP, that can be used to process FCS data[7].

The base dataset used in this analysis is the nine-color B-NHL panel published in our previous work.[8] Four additional MFC datasets were obtained that are used as the target datasets (see Flow cytometry data in the Methods for details). The model setup for TL is also detailed in the Methods.

Comparing merged and original datasets

To evaluate the quality of the merged dataset, we use Jensen-Shannon distance (JSD) to quantify the similarity between the distributions of markers in the original and merged datasets, resulting in values between 0 (identical distributions) and 1 (totally disjoint distributions). If p and q are the probability distributions of a marker in the original and merged data, then the JSD is calculated as the square root of Jensen-Shannon divergence[15]:

$$JSD = \sqrt{\frac{D(p||m) + D(q||m)}{2}},$$

where m is the pointwise mean of p and q, and D is the Kullback-Leibler divergence.[16]

We compute the JSD for each non-shared marker between the original and merged sample for all datasets. We obtained a mean JSD score of less than 0.1 for all the datasets, indicating good agreement between the merged and original datasets in terms of marker distribution. The individual JSD score for each non-shared marker and the average JSD for each dataset are reported in Figures S2 and S3. Although JSD scores do not

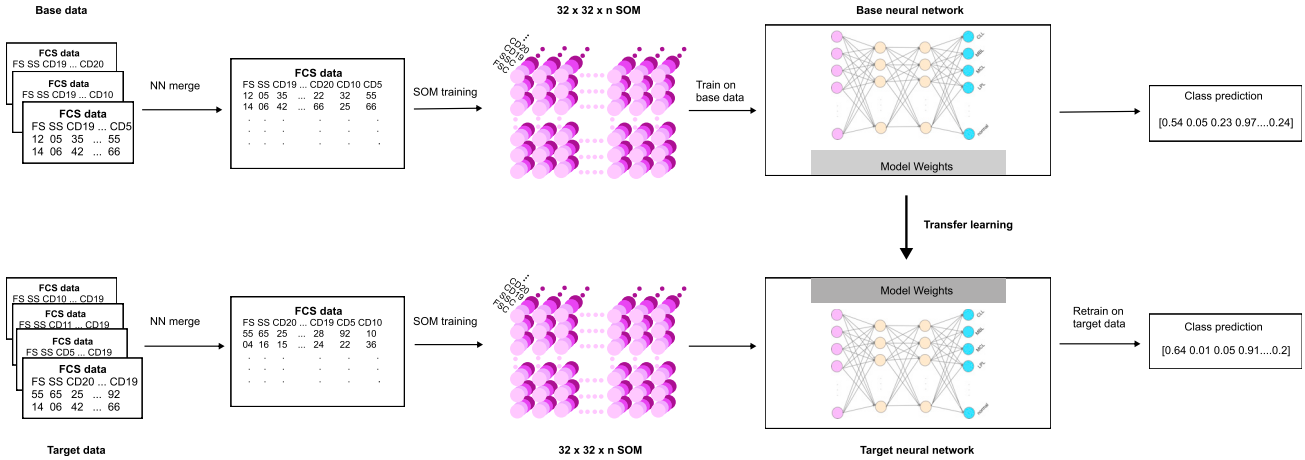


Figure 1. Overview of the knowledge transfer pipeline

For each dataset, FCS files from different tubes for each sample are merged using the NN merge. Next, individual SOMs are generated for each of the merged FCS samples. The SOM nodes are arranged in a 32 × 32 grid where each node is associated with an n-dimensional weight vector, where n is equal to the number of channels in the original FCS events. The SOM node weights are then used as input to the CNN. The weights from the base model trained on the base dataset are transferred to each of the target networks. The target networks are then retrained on the respective target dataset to generate class predictions

evaluate the extent to which a cell population (based on co-expression of markers) is preserved in the merged data, the scores provide a way of assessing how the imputation affected individual marker distribution. Furthermore, the original and merged models' performance for the base data were compared to evaluate the effect of NN merge on the CNN classification. The merged base model achieved an overall weighted F1 score of 0.94 and an average F1 score of 0.74. In comparison, the original model trained with the unmerged FCS data from tubes 1 and 2 achieved an overall weighted F1 score of 0.94 and an average F1 score of 0.75, indicating that the NN merge did not introduce significant artifacts that negatively impact the CNN classification.

Evaluation of knowledge transfer

To evaluate knowledge transfer, we compared the performance of the target models with and without TL. A 10-fold validation was performed on both the standalone and TL models for each target dataset. For each model, weighted and average F1 scores were calculated. The models with TL showed a significant improvement in F1 scores, especially the average F1 scores for all the datasets (Figure 2). The delta in the performance between the datasets may be attributed to the size of the dataset, the quality of the original data, and the quality of the merged data with imputed marker values. The effect of the quality of merged data on the classification score is discussed in Figure S3B. The overall scores obtained by averaging the F1 scores over the 10-fold validation and the 95% CI values for the four datasets are reported in Table 1. The ROC curves and mean AUC for standalone and TL models are shown in Figure S5.

Furthermore, we show that the TL models converge with the standalone models with the chosen parameters (Figure S7). The TL models have a lower initial validation loss and reach the asymptote faster than the standalone models. While the TL loss for the Erlangen panel does not converge with the standalone model, the classification performance is still improved with TL (Figure 2). The lack of convergence in terms of model loss could be a result of the different diagnostic setup in Erlangen, resulting in a small, highly imbalanced dataset that greatly diverges from the base data.

Learning curve analysis

Here, we describe two use cases that change the MFC diagnostic panel and require an AI model to be adapted. We use our current workflow to adapt the base model for both cases and analyze the model's learning curves for each case. A learning curve shows the model's score for varying numbers of training samples and can be used to compare different settings or algorithms and determine the amount of data used for training.[17] We demonstrate that TL with merge increases the models' overall performance and the models have a higher start on the learning curve for smaller sample sizes.

Case 1: Transition to a new cytometer within the same laboratory

In MFC diagnostics, switching to a device that supports more fluorochromes per measurement is a common transition in a diagnostic laboratory that optimizes its workflows by updating its equipment. Usually, this process involves a few weeks, during which samples are measured with both protocols, the old one validating the new one. However, this means that only a few samples from the new protocol are available to train a new classifier. Using knowledge transfer, we show that transition can be handled quickly by adapting an existing AI model. We set up a transition scenario from a five-color cytometer to nine-color using our MLL5F and MLL9F datasets. We trained a model with the MLL5F panel and used this as the base network to train a new model for the MLL9F panel. We used an increasing number of samples in the training set for each iteration of the learning curve, while the validation set for each iteration was kept the same. We started with 5 random samples per class and iteratively increased the number of training samples by 5 in each class until

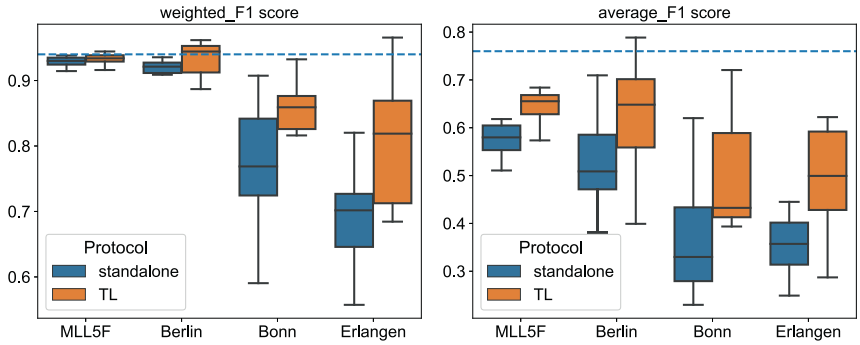


Figure 2. Performance for standalone versus transfer learning

The boxplots show F1 scores obtained: on the left, weighted_F1_scores are plotted for each dataset and, on the right, the average_F1_scores are shown. The blue dotted line across the plots represents the previously reported base model's performance, which is considered expert-level accuracy for this work. The transfer learning models perform better in all four datasets. These models achieve a higher F1 score, especially the average f1 score. A significant increase in average_F1 score is seen for MLL5F ($p = 1.805 \times 10^{-3}$) and Erlangen ($p = 3.194 \times 10^{-2}$) panels. For Bonn and Berlin panels, we achieved a p value of 6.838×10^{-1} and 1.659×10^{-1}, respectively. All p values were computed using an independent t test with Bonferroni correction.

50 random samples per class. F1 scores were recorded for each iteration. The learning curve (Figure 3A) with TL shows a higher start and asymptote for the target network; the confusion matrix obtained with five training samples per class (Figure S6A) shows a significant improvement in classification, especially for the smaller classes.

Case 2a: Model adaptability across laboratories

MFC diagnostic workflows are relatively similar across laboratories. However, the MFC panel used for diagnosis varies depending on the cytometer and antibodies measured. For an AI model, the reported performance is valid for the given MFC panel. When the model is used to interpret different MFC data, the performance drops significantly without changes to the underlying architecture and parameters. Training a new model requires a longer training time and large datasets. Here, we demonstrate that our workflow can extend a model trained on a specific MFC panel with an extensive training dataset to different MFC panels with lesser data (Table 2).

We used our merged base model (MLL9F_base) to train new models for Bonn and Berlin panels. Both target models showed a significant increase in overall performance with TL. As with the previous experiment, the learning curves were obtained for an increasing number of training samples in each class. The target models were trained with 5 random samples per class, which were gradually increased to 50 samples per class. The F1 scores showed a significantly higher start and overall performance in inter-laboratory adaptation with our workflow (Figures 3B and 3C).

Case 2b: Cross-laboratory adaptation with different diagnostic setting

A screening panel was used for the Erlangen dataset to diagnose B cell neoplasms with a separate classification panel for further subtype determination. We trained a model with the same architecture and parameters as our MLL9F_base model for the screening panel to obtain a "normal" versus "pathological" binary classification. The resulting model could classify 86% of pathological samples and 96% of normal samples correctly. Furthermore, we used our current workflow for the 247 samples (see Table 2) with both screening and the classification panel (B1 and B2). We employed knowledge transfer as described and saw an overall gain in the average F1 score from 0.33 to 0.52. The learning curve (Figure 3D) showed a higher start and asymptote, similar to the other three datasets.

DISCUSSION

The use of AI models in the diagnosis of hematological malignancies is steadily increasing over time.[18,19] Several AI algorithms have been developed to improve accuracy in lymphoma subtyping using high-throughput data such as MFC. While MFC data are ideal for deep learning applications, the protocol is not uniform between laboratories or within the same laboratory over time, leading to changes in the data. Thus, a model trained on a specific MFC protocol cannot be applied to a dataset with a different protocol.

This study presents a workflow to extend AI models trained on a specific MFC panel to multiple MFC panels and data sizes. Our workflow allows an existing model to adapt quickly to any changes in the data making it possible to be deployed in a routine diagnostic setting across different laboratories.

With our work, we demonstrate the application of TL to improve the performance and adaptability of AI to multiple datasets. We use knowledge from the base model trained on a specific MFC panel to train target models for new MFC data. Ideally, TL is applied in cases where the base and the target tasks are related yet different, whereas the datasets do not change in terms of composition. Our work shows that TL can be used successfully when the base and target datasets change, while both the base and target tasks remain the same.

Our proposed workflow combines knowledge transfer with FCS data merging (Figure 4). Merging multiple aliquots is a known approach for increasing computational depth for deep phenotyping and FCS analysis.[20] In the context of a CNN, it increases the network's feature space by combining markers measured in different tubes. It also allows us to maximize our networks' transferability, which is essential for a successful knowledge transfer.

We extend our base model to four additional datasets with a varying number of tubes per sample and markers with no changes to the model architecture and training parameters. Here, we show that knowledge transfer in conjunction with FCS data merging enhances the overall performance for target models by allowing for already learned features from a large dataset to be transferred to smaller and different datasets.

With our workflow, the target models achieve an overall performance close to the previously reported expert-level accuracy.[8] For the Berlin panel, the TL model achieved a median weighted

Table 1. Performance metrics

Protocol	Scores	MLL 5F	Berlin	Bonn	Erlangen
With_TL	f1_weighted (95% CI)	0.93 (0.92, 0.93)	0.93 (0.91, 0.95)	0.85 (0.81, 0.88)	0.80 (0.73, 0.87)
	f1_avg (95% CI)	0.64 (0.61, 0.66)	0.62 (0.54, 0.71)	0.50 (0.41, 0.59)	0.52 (0.40, 0.64)
	Precision	0.91	0.93	0.82	0.71
	Recall	0.92	0.93	0.83	0.76
Standalone	f1_weighted (95% CI)	0.92 (0.91, 0.93)	0.92 (0.90, 0.93)	0.76 (0.69, 0.83)	0.69 (0.63, 0.74)
	f1_avg (95% CI)	0.57 (0.54, 0.59)	0.52 (0.45, 0.59)	0.40 (0.26, 0.53)	0.35 (0.31, 0.40)
	Precision	0.90	0.92	0.75	0.58
	Recall	0.91	0.92	0.82	0.73

Weighted and average F1 score along with 95% confidence interval (CI) values for the four target datasets for models with knowledge transfer and standalone models without transfer learning are reported here. The F scores were calculated as an average of the 10-fold validation for each dataset. Precision and recall are calculated as the weighted average of the per class scores for each fold and then averaged.

F1 score of 0.94, the same as expert-level performance. This enhancement could only be achieved by combining FCS data merging with TL. While TL allows for features already learned to be transferred between models to enhance the overall performance of target models, merging multiple FCS tubes makes it possible to apply maximum TL between different MFC datasets.

Furthermore, the learning rate of target models with TL is much higher than the standalone models, as demonstrated by our learning curve analysis. The TL models achieve significantly higher performance for very small training sizes. In the context of transition to a new cytometer, this would allow an already deployed AI model to be quickly adapted to the new protocol without having to wait for a considerable time for enough samples to become available for the new protocol.

While the proposed workflow successfully allows the AI model to be adapted to different MFC data, it does not entirely address the inherent differences between various datasets. Each laboratory has a different diagnostic goal and expertise, leading to different panel designs and different data distribution among the classes for each dataset. The class imbalance within a given dataset can be accounted for in the CNN using appropriate class weights during training. However, these class weights are not transferrable, and thus the non-uniform imbalance between the various datasets cannot be addressed within the CNN. Advanced data augmentation strategies to artificially create more samples for the rare classes could allow for a uniform data distribution among the datasets and are currently being explored.

The choice of marker combinations used for each MFC panel depends on the diagnostic workflow and preferences of the laboratories. While some markers are standard markers for B cell neoplasm assessment, others are specific to certain subtypes, and different laboratories may use alternate markers for such cases. The differences in the marker combinations between the panels are addressed using NN merge and SOM training in our workflow. While marker alignment in SOM calculation accounts for overlapping CD markers between the base and target MFC panels, missing markers in the target datasets are handled by setting these values to zero in SOM weight calculation. These markers may be necessary for specific subtype identification in base data and could impact the classification of these subtypes in the target models. For instance, IgM, a marker that Munich chose to improve LPL (lymphoplasmacytic lymphoma) classification in the MLL9F panel, is missing in the Bonn and Erlangen panels. While these panels use other known markers, such as CD38 for LPL identification, the information contained in IgM might be lost during knowledge transfer and thus can impact the classification performance for this class. This might also explain the decline in performance for LPL, which can be seen in confusion matrices for the Erlangen panel (Figure S6D). Despite these inherent biases that can confound the classification performance, we see an overall performance enhancement for all four target sets with the proposed workflow.

While TL helps adapt and improve model performance, the result must be carefully evaluated for each case. Especially, evaluation on small and highly imbalanced datasets often encountered in the routine laboratory setting can cause misleading results without a thorough assessment of different performance aspects.

In conclusion, we present a workflow to extend deep learning models to multiple MFC panels and achieve high accuracy for multi-label classification across datasets. Here, we address some of the previous challenges for automated flow cytometry classification by allowing models to be trained with smaller training sizes and generalizing models to work with multiple MFC panels. Our workflow is a step toward making deep learning models robust so that AI for diagnostic MFC can move from the "proof of concept" stage into routine diagnostics.

Limitations of the study

This section discusses the limitation of our work in terms of known shortcomings of the merging approach, technical variance between datasets, and potential improvements. Although NN is a well-known method for data imputation, for merging FCS events, NN merging is known to sometimes introduce a spurious combination of markers into the imputation results.[21] However, this did not lead to a reduced performance of our classification model. Both merged and unmerged models produced nearly identical F1 scores for the base dataset. Furthermore, we also looked at the impact of the number of shared markers on the imputation quality and did not find any differences (see experimental procedures, SE1). While TL accounts for some of the variability between the datasets, the technical variation arising from sample preparation, equipment calibration cannot be completely ruled out and could potentially affect the classification performance. A standardized normalization approach

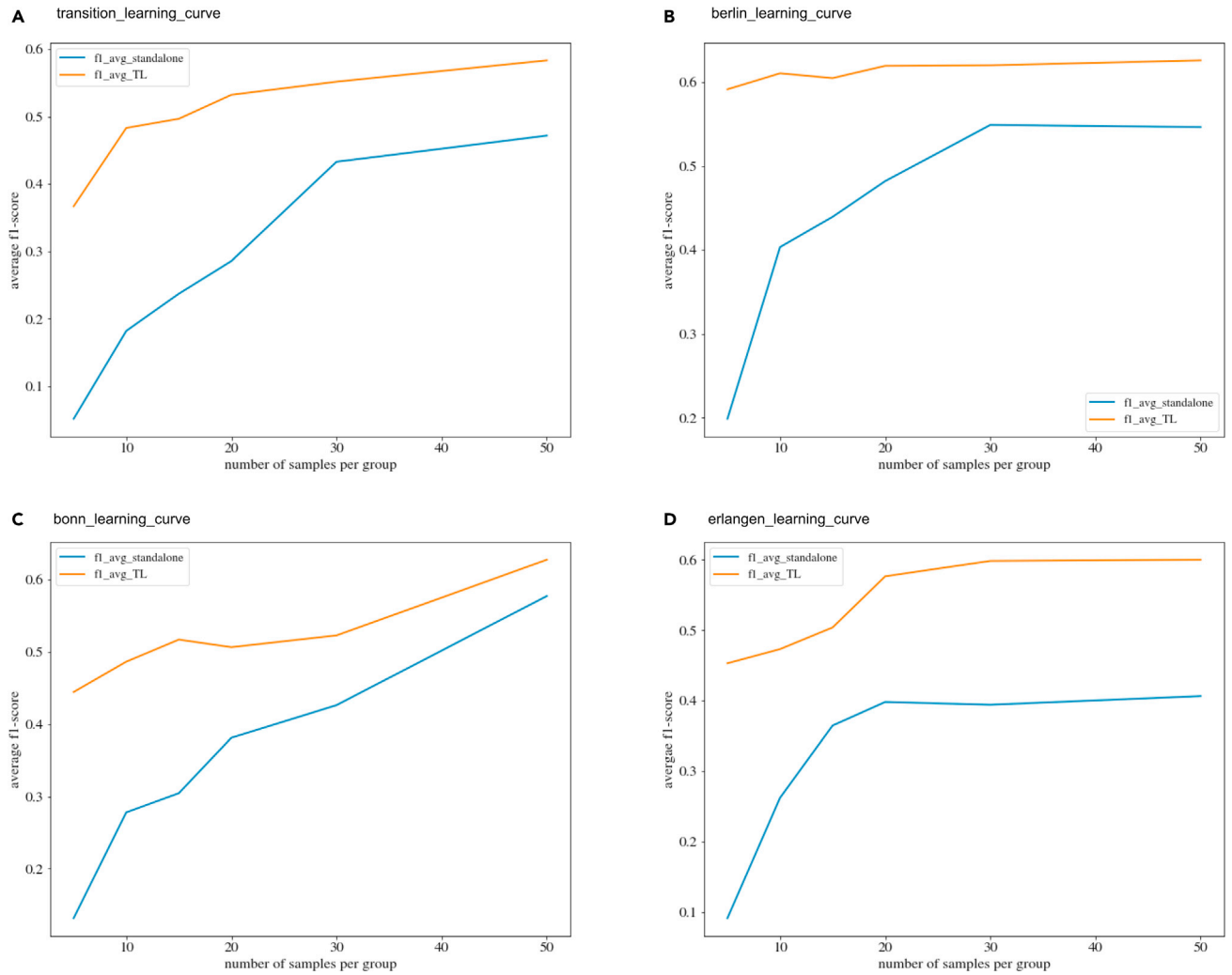


Figure 3. Learning curves

The learning curve for average f1 scores with various training sizes for all the four target datasets is shown here. The curves were obtained with randomly sampled training examples. We start with five training samples in each class and iteratively increase up to 50 samples per class. In cases where 50 samples are not available for a given class, existing samples are randomly resampled to create up to 50 samples for the learning curve analysis. The curve for the transition experiment is shown in (A), while the curves for cross-laboratory experiments with Berlin, Bonn, and Erlangen panels are shown in (B), (C), and (D), respectively. The learning curves for all panels show a higher start and asymptote with transfer learning and an overall performance enhancement.

across datasets could improve the classification performance further. Although, this would add considerable computational overhead and may require a reference sample to be analyzed across various locations that can be used to remove all the technical variation. Finally, we align FCS channels by matching CD markers while ignoring the fluorochromes for our knowledge transfer. While any missing markers are handled within the updated SOM training, the current workflow will ignore new markers. The information lost because of the marker alignment could impact the classification of specific subtypes and the overall performance. The performance may be improved further with partial knowledge transfer techniques, where features from existing channels are transferred while the model is trained to learn the new channels present in the new protocol.[22] All five of the datasets used in this study are from Navios cytometers. Although the workflow presented here is not limited to datasets acquired on a specific device, our models could have a potential vendor bias that should be considered when data are acquired on a device from a different vendor.

EXPERIMENTAL PROCEDURES

Resource availability

Lead contact

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Peter Krawitz (pkrawitz@uni-bonn.de).

Materials availability

There are no physical materials associated with this study.

Data and code availability

- All five FCS datasets are available at Harvard Dataverse: https://doi.org/10.7910/DVN/CQHHEH.
- The source code for merging FCS files, model generation, and transfer learning is available under an open-source license on git repositories.

Access details for data and code can be found here: https://flowcat.genetalk.de.

- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

Ethics approval

IRB or ethics approval does not apply as the study was conducted on fully anonymized retrospective patient data. A waiver was granted by the University of Bonn Medical Faculty Ethics Committee.

Methods

Flow cytometry data

The base dataset consists of around 18,000 training samples acquired using a 9-color MFC panel at Munich Leukemia Laboratory (MLL) between 2017 and 2018. Four additional MFC target datasets were obtained with different MFC panel compositions. The number of samples per cohort in each of the four target datasets are summarized in Table 2.

Munich five-color panel. A 5-color panel consisting of 10,079 samples was acquired at MLL between January 1, 2011, and December 31, 2012. For the assessment of B cell neoplasms (B-NHL), a panel consisting of seven 5-color combinations of monoclonal antibodies was used in all samples to analyze the surface expression of 20 antigens. A detailed antibody-color combination is given in Table S1A. We refer to this panel as the MLL5F panel.

Bonn nine-color panel. The second dataset was obtained from the University Hospital Bonn, consisting of 525 samples measured between January 1, 2018, and December 31, 2018. A panel composed of two 9-color combinations of monoclonal antibodies was used to analyze 16 antigens' surface expression for B-NHL assessment. Detailed MFC panel information is given in Table S1B. We refer to this panel as the Bonn panel in this work.

Berlin eight-color panel. For the third dataset, an 8-color panel consisting of 2,773 routine diagnostic samples from patients with suspected B cell neoplasms analyzed between January 1, 2016, and December 31, 2018, was obtained from the Berlin Hematology laboratory. The B-NHL assessment panel consisted of four 8-color combinations of monoclonal antibodies. Table S1C details the MFC panel used. In the following, we refer to this panel as the Berlin panel.

Erlangen panel. A fourth target dataset was obtained from the University Hospital Erlangen. The dataset consisted of 1,626 routine diagnostic samples from patients with suspected B-NHL analyzed between January 1, 2014, and July 31, 2020. The assessment panel consisted of a screening panel (B1), with one ten-color combination of monoclonal antibodies used to analyze the surface expression of nine antigens. Next, a secondary panel (B2) to identify the B-NHL subtype was used where necessary. Finally, for the identification of HCL (hairy cell leukemia), a third panel (B3) was used. All three panels are described in detail in Table S1D. For this study, we only consider the 247 samples with both B1 and B2 panels.

All samples were analyzed on Navios cytometers (Beckman Coulter, Miami, FL). Information on the number of events acquired for each panel is described in Table S2. All diagnoses were verified with additional tests from histology, cytomorphology, and *in situ* fluorescence hybridization, and only cases with unambiguous labels were used to train the models. Furthermore, only samples obtained from peripheral blood or bone marrow aspirate were included in the analysis. Flow cytometry data are stored in the FCS 2.0 format.[23] The compensated FCS 2.0 data segment has been used in the analysis.

Merge overview

The merge process is depicted in Figure 4. The steps for matching events between different data files are as follows.

Step1: determine the shared markers for each of the datasets (Table S3). The shared markers are used as the vector to calculate the distance between events in different data files.

Step2: take tube *i* (start with the first tube; $i = 1$), and iterate over all of the remaining tubes *j*.

Step3: for each event in tube *i*, calculate the NN in tubes *j*.

Step4: copy the tube-specific marker (non-shared marker) values from the computed NNs in tube *j* to the events in tube *i*.

Step5: increment *i*, repeat the above steps. Events in each tube will now have imputed values for markers that were measured in a different tube.

Table 2. B cell lymphoma cohorts in target datasets

	MLL 5F	Berlin	Bonn	Erlangen
CLL	1,886	420	96	72
MBL	221	–	–	16
MCL	102	50	12	21
PL	151	–	–	–
LPL	121	3	6	9
MZL	40	15	5	10
FL	120	49	20	10
HCL	259	54	13	2
Normal	5,836	2,182	404	107

Data distribution among the different cohorts for each dataset is shown here. Only data samples with precise diagnoses were included. For the Erlangen panel, only samples with both B1 and B2 panels are shown. CLL and MBL are merged into a single class for classification. CLL, chronic lymphocytic leukemia; HCL, hairy cell leukemia; FL, follicular lymphoma; LPL, lymphoplasmacytic lymphoma; MBL, monoclonal B cell lymphocytosis; MCL, mantle cell lymphoma; MZL, marginal zone lymphoma; PL, prolymphocytic leukemia.

Step6: merge all the events across all tubes into a single large matrix.

The resulting data file obtained after merging the original data files and calculating each event's values was a file containing information about all parameters measured in all multicolor staining for each of the events recorded. For the MLL5F panel, we merge tubes 2, 3, 4, 5, and 7. Thus, each merged/calculated data file contained all 18 parameters measured for each of the $\geq 2.5 \times 10^5$ events analyzed per sample (5 aliquots/sample $\times \geq 5 \times 10^4$ events/aliquot). The tubes merged for the different datasets and the merge parameters are described in Table S3. We implement the merge using scikit-learn API.[24]

SOM

A SOM is a network of interconnected nodes, ordered in a two-dimensional topology, which can be used for unsupervised clustering of high-dimensional data.[25] SOMs were used as a method to reduce the dimensionality of the data while preserving its spatial structure. We first generate individual SOMs for each of the merged FCS samples. Each node in the SOM is associated with a "weight" vector representing the n-dimensional FCS data. Individual SOM transformation uses pre-initialized node weights from a reference SOM. A reference SOM from the base model is used as the pre-initialized weights for the target datasets to ensure the same initial tree structure. Markers are aligned to the base dataset by matching FS, SS, and as many CD markers as possible by disregarding the associated fluorochromes. In the case of missing markers in the target set, they are set to "n/a"; any new CD markers in the target set that are not found in the base set are ignored. The SOM implementation was adapted to account for missing data values by modifying the training process.[26] We introduce a masking matrix with values in {0, 1} for each value in the original data: "1" indicates that the data value is valid and "0" indicates that the data value is invalid, and the data point should be ignored for any calculations. The SOM training is then adjusted to use the mask values to ignore invalid data points for the best-matching unit calculation and weight updates.

CNN

System requirements. The SOM generation and classification both require Tensorflow.[27] An NVIDIA GPU is preferable for running all computations. We used a Tesla P40 GPU with 24 GB GDDR5X memory on an Ubuntu 16.04 Linux machine with TensorFlow 1.12. In addition, at least 500 GB HDD storage for the entire dataset is necessary. The computation time required for analysis depends on the size of the dataset. For our largest dataset, the SOM generation for merged files took approximately 30 h, and the CNN training required an hour. If video RAM is a limitation, reducing the batch size for the SOM generation might be beneficial.

Datasets. We used the nine-color panel dataset from our previous model as the base dataset, referred to as the MLL9F panel. The four additional panels

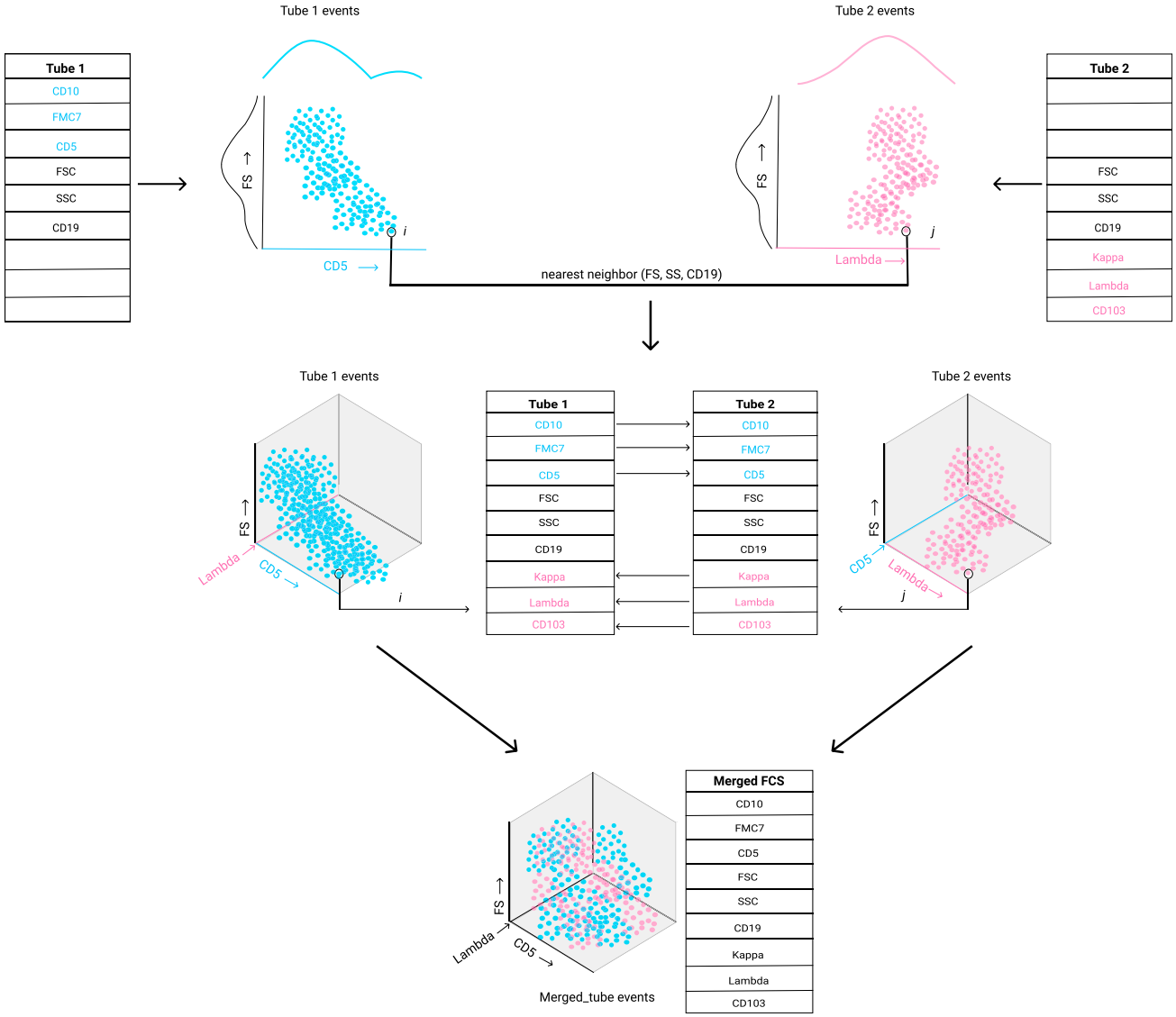


Figure 4. Merge overview

Overview of NN merge is shown here for two tubes with three shared markers. Each tube has three tube-specific markers: CD10, FMC7, and CD5 are tube 1-specific markers, while tube 2-specific markers are Kappa, Lambda, and CD103. Events are shown in a two-dimensional space with one shared marker (FS) and one tube-specific marker (CD5 for tube 1 and Lambda for tube 2). For each event "*i*" in tube 1, the NN in tube 2, "*j*," is computed in terms of the shared markers (FS, SS, and CD19). Next, tube 2-specific markers from "*j*" are copied over to "*i*"; tube 1-specific markers from "*i*" are copied over to "*j*." The process is repeated for all the events in tubes 1 and 2. All the tube 1 events will have imputed values for tube 2-specific markers (Kappa, Lambda, and CD103) and tube 2 events will have imputed tube 1-specific markers (CD5, CD10, and FMC7); these events can now be analyzed for the imputed markers that were previously missing. Finally, the expression vectors of all events across tubes are merged, resulting in a combined FCS file consisting of events from both tubes with all the measured parameters.

described above: MLL5F, Bonn, Berlin, and Erlangen panels, are the target datasets on which TL is applied and evaluated.

Model setup. The modified CNN architecture for the merged data is shown in Figure S4. The model generates predictions using SOM node weights for a number of classes. The weights are first processed with three convolution layers with decreasing filter sizes, followed by a global max-pooling layer that summarizes filters across the SOM map's spatial dimension. Next, two dense layers combine information, and a final softmax layer generates class predictions. Models are trained using Adam optimizer[28] with a learning rate of 0.001. A global weight decay of 5×10^{-6} was applied to all layers. The model is implemented in Keras.[29]

We trained a base model for the merged base dataset with the modified CNN architecture, referred to as MLL9F_base. Two models were trained for each target dataset: a standalone model without knowledge transfer and a second model with knowledge from the base model (MLL9F_base). The weights for each layer in the target model with TL are initialized with trained weights from the base model's corresponding layer, while for the standalone models these are randomly initialized. The standalone models' hyperparameters are kept identical to the base model—20 epochs, a learning rate of 0.001, and a global decay of 5×10^{-6}. For the second set of models with TL, we used the same learning rate and global decay while the number of epochs was reduced to 15. Furthermore, the two dense layers are frozen by

setting the "trainable" hyperparameter as false. When using TL, the norm is to freeze the convolution layers and retrain only the dense layers to avoid overfitting. However, in our case, the MFC panel composition is different from the base data. Therefore, to account for changes in the panel, we keep the convolution layers unfrozen and retrain them to learn the filters for the target MFC panel. Instead, we freeze the two dense layers that combine information for generating class prediction since the classes to be predicted are the same as in the base task.

We perform 10-fold validation for all the target datasets to avoid any bias resulting from a single random train-validation split, especially for the smaller datasets. Each target model is trained on the training split, and performance metrics are calculated for the validation split of the respective target dataset. The average scores across the 10-fold validation are reported as the final performance measure.

Performance metrics. Precision and recall per class were defined on the true label of each case. Prediction performance was evaluated using F1 scores. The F1 score is the harmonic mean between recall and precision and places equal importance on both measures. We use the F1 score as a performance metric to reflect the real-world diagnostic scenario where precision and recall are equally important. The average per class F1 scores was calculated as

$$avg\ f_1 = \frac{1}{|C|}\sum_{c\in C} f_c,\ with$$

$$f_c = 2\,\frac{Precision_c \cdot Recall_c}{Precision_c + Recall_c}\ where\ C\ is\ the\ set\ of\ all\ classes.$$

The weighted F1 score was calculated as the class-size-weighted average of the per class F1 scores

$$weighted\ f1 = \frac{1}{\sum_{c\in C} s_c}\sum_{c\in C} s_c f_c\ with\ s_c\ as\ the\ number\ of\ cases\ in\ class\ c.$$

For each model, we calculate top 1 accuracy rate of the classifier for the eight classes: chronic lymphocytic leukemia and its predecessor monoclonal B-cell lymphocytosis (CLL/MBL), marginal zone lymphoma (MZL), mantle cell lymphoma (MCL), prolymphocytic leukemia (PL), follicular lymphoma (FL), hairy cell leukemia (HCL), and lymphoplasmacytic lymphoma (LPL) and healthy controls. MBL is a diagnostic finding that is regarded as a potential preneoplasia and precursor of CLL in most cases.[30] Both MBL and CLL, therefore, share a similar immunophenotype (CD5+/CD19+/CD20 low/CD23+/Ig low). Therefore, we combine MBL and CLL into a single class for classification. However, we list MBL as a separate class in the learning and ROC curves for a fine-grained analysis of classification sensitivity.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.patter.2021.100351.

ACKNOWLEDGMENTS

This study was supported by the German Research Council (Deutsche Forschungsgemeinschaft [DFG]), by project 315041274, awarded to P.M.K.

AUTHOR CONTRIBUTIONS

Conceptualization, N.M., M.Z., and P.M.K.; methodology, N.M. and M.Z.; software, N.M. and M.Z.; visualization, N.M. and M.Z.; formal analysis, N.M., M.Z., L.M., A.H., F.E., and H.L.; writing – original draft, N.M. and M.Z.; writing – review & editing, N.M., M.Z., P.M.K., and S.W.K.; data curation, N.M., M.Z., and L.M.; funding scquisition, P.M.K.; resources, T.H., W.K., J.W., P.B., and S.W.K.; supervision, P.M.K.

DECLARATION OF INTERESTS

H.L. is a founder and employee of res mechanica; F.E. is an employee of res mechanica; W.K. is a founder and employee of MLL; T.H. is a founder and employee of MLL.

Received: April 12, 2021
Revised: May 10, 2021
Accepted: August 25, 2021
Published: September 17, 2021

REFERENCES

1. Shapiro, H.M. (2003). Practical Flow Cytometry (Wiley-Liss).
2. Henel, G., and Schmitz, J.L. (2007). Basic theory and clinical applications of flow cytometry. Lab. Med. *38*, 428–436.
3. Craig, F.E., and Foon, K.A. (2008). Flow cytometric immunophenotyping for hematologic neoplasms. Blood *111*, 3941–3967.
4. Bendall, S.C., and Nolan, G.P. (2012). From single cells to deep phenotypes in cancer. Nat. Biotechnol. *30*, 639–647.
5. O'Neill, K., Aghaeepour, N., Špidlen, J., and Brinkman, R. (2013). Flow cytometry bioinformatics. Plos Comput. Biol. *9*, e1003365.
6. Matek, C., Schwarz, S., Spiekermann, K., and Marr, C. (2019). Human-level recognition of blast cells in acute myeloid leukaemia with convolutional neural networks. Nat. Mach. Intell. *1*, 538–544.
7. Ng, D.P., and Zuromski, L.M. (2020). Augmented human intelligence and automated diagnosis in flow cytometry for hematologic malignancies. Am. J. Clin. Pathol. *155*, 597–605.
8. Zhao, M., Mallesh, N., Höllein, A., Schabath, R., Haferlach, C., Haferlach, T., Elsner, F., Lüling, H., Krawitz, P., and Kern, W. (2020). Hematologist-level classification of mature B-cell neoplasm using deep learning on multiparameter flow cytometry data. Cytom. Part A. *97*, 1073–1080.
9. Van Dongen, J.J.M., Lhermitte, L., Böttcher, S., Almeida, J., Van Der Velden, V.H.J., Flores-Montero, J., Rawstron, A., Asnafi, V., Lécrevisse, Q., Lucio, P., et al. (2012). EuroFlow antibody panels for standardized n-dimensional flow cytometric immunophenotyping of normal, reactive and malignant leukocytes. Leukemia *26*, 1908–1975.
10. Weiss, K., Khoshgoftaar, T.M., and Wang, D.D. (2016). A survey of transfer learning. J. Big Data *3*, 9. https://doi.org/10.1186/s40537-016-0043-6.
11. Pedreira, C.E., Costa, E.S., Barrena, S., Lecrevisse, Q., Almeida, J., Van Dongen, J.J.M., and Orfao, A. (2008). Generation of flow cytometry data files with a potentially infinite number of dimensions. Cytom. Part A. *73*, 834–846.
12. Abdelaal, T., Höllt, T., Van Unen, V., Lelieveldt, B.P.F., Koning, F., Reinders, M.J.T., and Mahfouz, A. (2019). CyTOFmerge: integrating mass cytometry data across multiple panels. Bioinformatics *35*, 4063–4071.
13. Costa, E.S., Pedreira, C.E., Barrena, S., Lecrevisse, Q., Flores, J., Quijano, S., Almeida, J., Del Carmen García- Maclas, M., Bottcher, S., Van Dongen, J.J.M., et al. (2010). Automated pattern-guided principal component analysis vs expert-based immunophenotypic classification of B-cell chronic lymphoproliferative disorders: a step forward in the standardization of clinical immunophenotyping. Leukemia *24*, 1927–1933.
14. O'Neill, K., Aghaeepour, N., Parker, J., Hogge, D., Karsan, A., Dalal, B., and Brinkman, R.R. (2015). Deep profiling of multitube flow cytometry data. Bioinformatics *31*, 1623–1631.
15. Naghshvar, M., Javidi, T., and Wigger, M. (2015). Extrinsic Jensen–Shannon divergence: applications to variable-length coding. IEEE Trans. Inf. Theor. *61*, 2148–2164.
16. Kullback, S., and Leibler, R.A. (1951). On information and sufficiency. Ann. Math. Stat. *22*, 79–86.
17. Meek, C., Thiesson, B., and Heckerman, D. (2002). The learning-curve sampling method applied to model-based clustering. J. Mach. Learn. Res. *2*, 397–418.
18. Radakovich, N., Nagy, M., and Nazha, A. (2020). Artificial intelligence in hematology: current challenges and opportunities. Curr. Hematol. Malig. Rep. *15*, 203–210.

19. Shouval, R., Fein, J.A., Savani, B., Mohty, M., and Nagler, A. (2021). Machine learning and artificial intelligence in haematology. Br. J. Haematol. *192*, 239–250.

20. Robinson, J.P., Durack, G., and Kelley, S. (1991). An innovation in flow cytometry data collection and analysis producing a correlated multiple sample analysis in a single file. Cytometry *12*, 82–90.

21. Lee, G., Finn, W., and Scott, C. (2011). Statistical file matching of flow cytometry data. J. Biomed. Inform. *44*, 663–676.

22. Hassan, A. (2019). Transfer learning from RGB to multi-band imagery. Azavea https://www.azavea.com/blog/2019/08/30/transfer-learning-from-rgb-to-multi-band-imagery/.

23. Dean, P.N., Bagwell, C.B., Lindmo, T., Murphy, R.F., and Salzman, G.C. (1990). Introduction to flow cytometry data file standard. Cytometry *11*, 321–322.

24. Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., et al. (2013). {API} design for machine learning software: experiences from the scikit-learn project. In ECML PKDD Workshop: Languages for Data Mining and Machine Learning, pp. 108–122.

25. Kohonen, T. (1990). The self-organizing map. Proc. IEEE *78*, 1464–1480.

26. Samad, T., and Harp, S.A. (1992). Self-organization with partial data. Netw. Comput. Neural Syst. *3*, 205–212. https://doi.org/10.1088/0954-898X/3/2/008.

27. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). TensorFlow: a system for large-scale machine learning. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016.

28. Kingma, D.P., and Ba, J.L. (2015). Adam: a method for stochastic optimization. In 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings.

29. Chollet, F. (2015). Keras. Online https://keras.io.

30. Swerdlow, S.H., Campo, E., Pileri, S.A., Lee Harris, N., Stein, H., Siebert, R., Advani, R., Ghielmini, M., Salles, G.A., Zelenetz, A.D., et al. (2016). The 2016 revision of the World Health Organization classification of lymphoid neoplasms. Blood *127*, 2375–2390.