REVIEW ARTICLE

CYTOMETRY
Journal of Quantitative Cell Science PART A

MIFlowCyt

A Cancer Biologist's Primer on Machine Learning Applications in High-Dimensional Cytometry

Timothy J. Keyes,[1,2] Pablo Domizi,[2] Yu-Chen Lo,[2] Garry P. Nolan,[3] Kara L. Davis[2*]

[1]Medical Scientist Training Program, Stanford University School of Medicine, Stanford, California

[2]Department of Pediatrics, Stanford University School of Medicine, Stanford, California

[3]Department of Microbiology and Immunology | Baxter Laboratory for Stem Cell Biology, Stanford University School of Medicine, Stanford, California

Received 7 December 2019; Revised 10 March 2020; Accepted 12 May 2020

Grant sponsor: National Cancer Institute, Grant number: 1F31CA239365-01; Grant sponsor: NIH, Grant number: U54CA209971, Grant number: U54HG010426, Grant number: U19AI100627; Grant sponsor: The Parker Institute for Cancer Immunotherapy

Additional Supporting Information may be found in the online version of this article.

*Correspondence to: Kara L. Davis, Lokey Stem Cell Research Building (SIM1), 265 Campus Drive, Room G2078, Palo Alto, CA 94305
Email: kardavis@stanford.edu

Published online 30 June 2020 in Wiley Online Library (wileyonlinelibrary.com)

DOI: 10.1002/cyto.a.24158

Abstract

The application of machine learning and artificial intelligence to high-dimensional cytometry data sets has increasingly become a staple of bioinformatic data analysis over the past decade. This is especially true in the field of cancer biology, where protocols for collecting multiparameter single-cell data in a high-throughput fashion are rapidly developed. As the use of machine learning methodology in cytometry becomes increasingly common, there is a need for cancer biologists to understand the basic theory and applications of a variety of algorithmic tools for analyzing and interpreting cytometry data. We introduce the reader to several keystone machine learning-based analytic approaches with an emphasis on defining key terms and introducing a conceptual framework for making translational or clinically relevant discoveries. The target audience consists of cancer cell biologists and physician-scientists interested in applying these tools to their own data, but who may have limited training in bioinformatics.

Key terms

machine learning; mass cytometry; cancer; computational cytometry; data science

SINGLE -cell cytometry has proven to be a robust and flexible tool in both research and clinical laboratories since the early 1970s (1). In recent years, the advent of high-dimensional fluorescence cytometry (2), mass cytometry (3), and sequence-based cytometry (4) in particular have allowed for the generation of richer, more complex single-cell data than ever before. For basic scientists, these innovations have provided low-cost, high-throughput methods of profiling the phenotypic and functional characteristics of millions of individual cells across a wide array of human tissues. For clinicians, cytometry has become a mainstay of diagnosing (and guiding the treatment of) numerous medical conditions including infection, malignancy, and immunodeficiency (5). While much of the cytometry field has focused on characterizing the identity of and relationships between subpopulations of immune cells, an emerging field of particular significance is the application of high-dimensional cytometry to the study of cancer cell diversity and heterogeneity, which are difficult to characterize without many multiparameter, single-cell observations (6).

Historically, cytometry data have been analyzed "manually" via direct inspection of two-dimensional biaxial plots and the sequential application of Boolean gates that are hand-drawn based on the marker intensity distributions of individual cells (7). While this approach is familiar to many biologists and clinicians, it suffers from significant limitations—including high between-individual user bias, lack of scalability in the face of many markers or many individual samples, and a dependence on a priori knowledge regarding which cell populations are important for the biological question at hand (8). Importantly, these limitations are especially cumbersome in the study of cancer cells, whose segmentation into cellular subpopulations is generally less defined (and far more contentious) than that of healthy cells, which can

often be divided into discrete lineages relatively easily based on cell-surface marker expression.

In large part due to the limitations of manual gating-based analytic approaches, it is becoming increasingly common to analyze single-cell cytometry data using high-dimensional computational tools. In particular, the application of machine learning algorithms to cytometry data sets has increased significantly in the past 20 years, as has the application of artificial intelligence to biomedical data sets in general (Fig. 1). Many machine learning approaches have been recently adapted specifically for the analysis of cytometry data and have been shown to perform at least as well (and often better) than human experts on a variety of tasks (8, 9). Yet, despite the fact that these tools now exist, they are often nontrivial to understand and utilize to their full potential for most cancer biologists—and certainly clinicians—due to their stark departure from traditional manual gating workflows (10, 11). Similarly, machine learning analyses are often too complex for direct use in a clinical environment or require significantly larger data sets than are available to practicing physicians. Together, these issues demonstrate the difficulty of bridging the gap between data science and cancer systems biology in order to use cytometry data to answer important clinical or translational questions (12).

Here, we describe the main machine learning algorithms that have been used to analyze high-dimensional cytometry data in cancer biology, with an emphasis on what kinds of translational insights each of them can yield for the user. In doing so, we present the reader with a practical workflow for analyzing cytometry data by first starting with more exploratory, unsupervised machine learning approaches before working toward more targeted analytical methods. The primary audience for this review is cancer cell biologists and physician-scientists interested in applying machine learning algorithms to cytometry data in a clinically focused way, but who may have little to no bioinformatics background. Thus, what we present here is not meant to be an exhaustive guide, but rather a primer that will orient the reader and lead them toward relevant, in-depth, and up-to-date resources for further learning.

An Overview of Machine Learning and High-Dimensional Cytometry

We use the term "machine learning" here to refer to a broad range of computational techniques that involve training an arbitrary model to find, classify, or predict patterns in data according to a carefully selected set of rules (13). While some data scientists explicitly distinguish between traditional statistical models (such as linear or logistic regression) and more complex procedures such as building artificial neural networks (NNs) or conducting clustering analyses, we deliberately avoid this distinction here in order to provide a broad discussion of as many of the currently available tools as possible. Specifically, we give close consideration to three kinds of data analysis: dimensionality reduction, clustering, and predictive modeling (with feature selection), each of which have been successfully applied to cytometry data sets in cancer research. Importantly, each of these analytic strategies yield distinct insights and, in turn, are associated with specific input and output data formats that are critical for them to be used effectively by an investigator.

Dimensionality reduction and clustering are two forms of "unsupervised" machine learning. Unsupervised machine learning algorithms seek to describe how data are organized—either along a continuum or within distinct groups or clusters—based solely on the measurements associated with each observation. In the case of cytometry data, these measurements can correspond to a cell's transcript or protein expression levels, readouts of its genomic or epigenomic status, and/or information about its morphological or higher level spatial features (14, 15). Using these measurements, dimensionality reduction algorithms project the data into a lower dimensional (generally two- or three-dimensional) space in a way that preserves as much of the original information as possible and that can be easily visualized (7). Somewhat similarly, clustering algorithms increase the ease of visualizing and interpreting high-dimensional data by explicitly partitioning each observation—or, in the case of cytometry data, each cell—into discrete groups based on their similarity to one another. This allows cluster size and characteristics to be compared across multiple samples, experimental conditions, or treatment groups such that novel cell types of interest can be identified and characterized (10).

In contrast to dimensionality reduction and clustering, predictive modeling (of which there are many kinds) represents a form of "supervised" machine learning. Supervised machine learning relates the measurements associated with each observation in a data set to the corresponding value of an outcome variable of some kind. In cancer biology, outcome variables of interest are generally diagnostic or prognostic in nature, such as a patient's cancer type or subtype, their response to a specific therapy or therapies, or their likelihood of disease recurrence following remission (16). In addition to their direct application as predictive tools, supervised models are often informative because many of them can perform "feature selection," a process in which only the most important features for predicting a particular outcome are identified and included in the final model (17). Often, these selected features are biologically informative and can represent candidate therapeutic targets, molecular mechanisms of disease, or biomarkers in the diagnosis or surveillance of a particular cancer (18).

Together, dimensionality reduction, clustering, and predictive modeling can be used to guide a first-pass analysis of high-dimensional cytometry data by respectively allowing for easy data visualization, characterization of distinct cellular subpopulations, and selection of candidate features that most strongly predict the clinical or experimental outcomes of interest for the study at hand (Fig. 2). This framework, while by no means exhaustive, is explicit in its emphasis on focusing the analysis toward clinically useful insights. In general, clinical cytometry analyzes fewer simultaneous parameters than research cytometry, with clinical cytometers utilizing

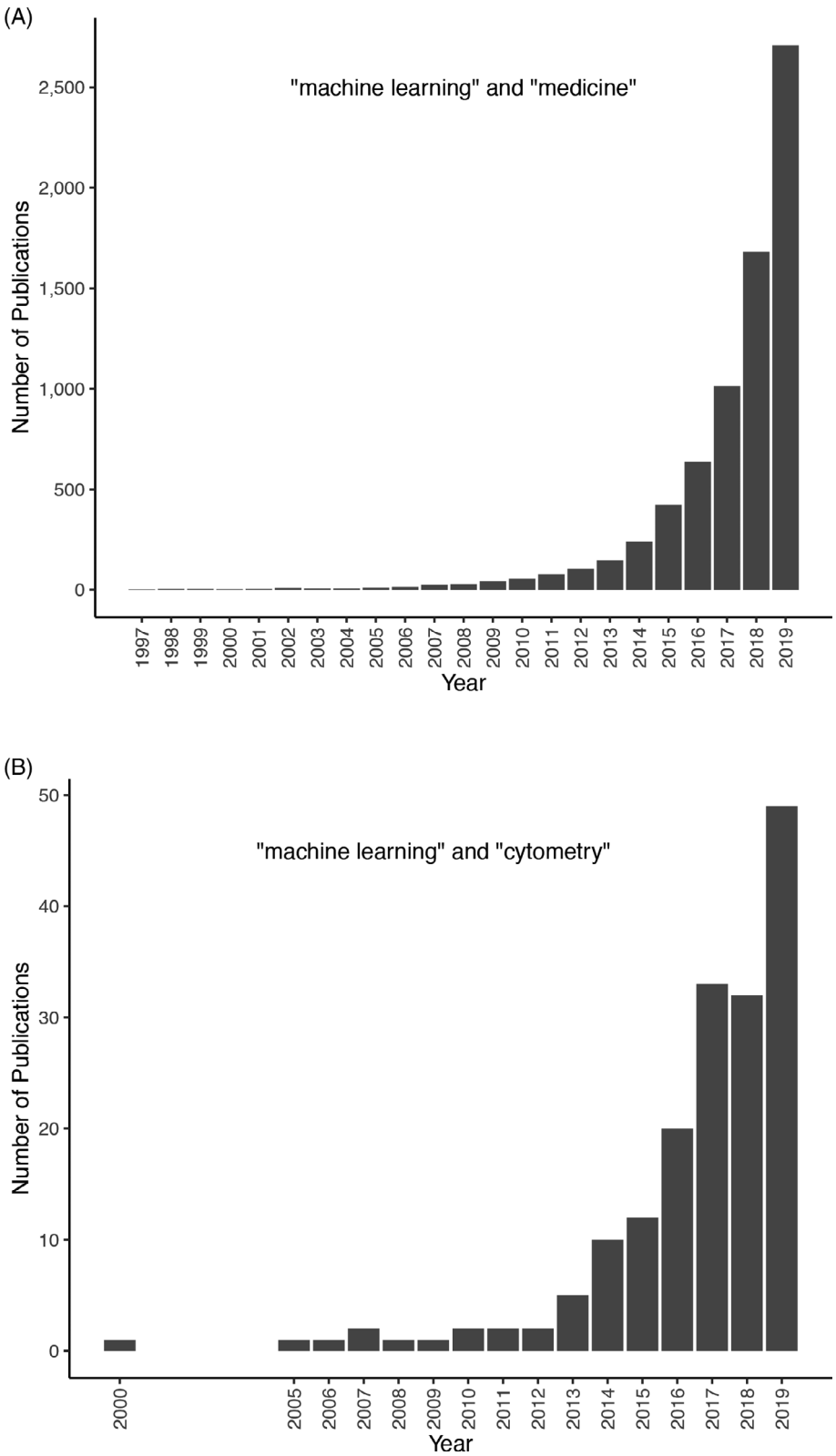

Fig 1. An increasing number of studies are using machine learning to analyze biomedical data. Bar graphs indicating the number of PubMed central search results for (**A**) the query "*machine learning*" and "*medicine*" since 1997 and (**B**) the query "*machine learning*" and "*cytometry*" since 2000.

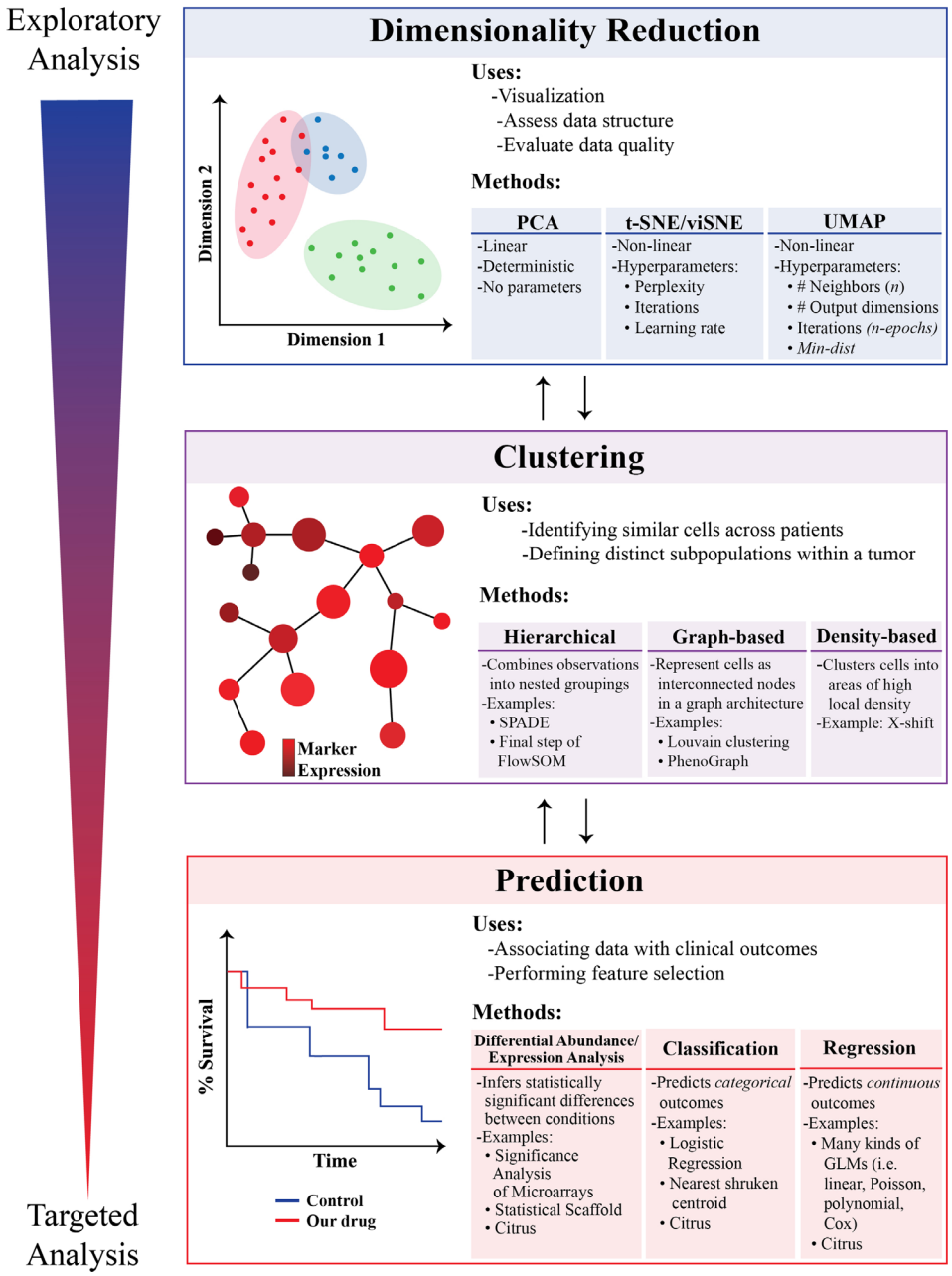

Fig 2. Schematic *diagram representing the analysis strategy described in this review. We encourage the reader to begin their analyses using exploratory approaches such as dimensionality reduction and basic visualization, later progressing to unsupervised clustering and predictive modeling/correlative biology. Bidirectional arrows are included in the diagram to emphasize that each stage of an analysis will influence multiple other stages, with results from earlier stages often informing the analytic approach in the subsequent stage (and vice versa). Throughout the figure, exploratory analyses are coded as blue, whereas more targeted analyses are coded as red.*

only two to eight colors on average for a given diagnostic test (19). For cancer specifically, flow cytometry is generally used in several well-defined ways: (a) quantifying the presence of rare, aberrant cell populations such as in the detection of minimal residual disease (MRD) in the surveillance of leukemia (20), (b) measuring the expression level of a particular protein or combination of proteins within a patient's tumor cells (21), or (c) identifying the clonality or lineage status of a patient's bulk tumor population relative to a reference control (22).

Given these restrictions on clinical cytometry in practice, it makes sense to organize translational cytometry studies around identifying a small number of specific surface markers, intracellular signals, or cell populations that could be potentially measured in a clinical lab. Organizing one's analysis first on dimensionality reduction, then on clustering,

and then on predictive modeling is one way of doing so (23). The following sections outline each part of this three-step workflow in closer detail while providing examples from the literature of how each of these strategies have been successfully applied to cancer cytometry data. In addition, glossary defining several key terms appears in the supporting information.

Before You Start: Quality Control and Data Preprocessing

Some readers will be familiar with the phrase "garbage in, garbage out" that is often used to describe a frustrating reality in the field of artificial intelligence: that is, if low-quality data are provided to even the best machine learning algorithm, it is unlikely that the results will be robust, reproducible, or even interpretable in a meaningful way (24). In the specific context of high-dimensional cytometry data, this means that data sets collected using flawed experimental design practices or in which technical artifacts have not been corrected are likely to produce spurious findings when analyzed with any machine learning tool. Furthermore, it is often more difficult to detect such errors after complex analyses have been performed. For these reasons, the importance of quality control and data preprocessing cannot be overstated. A full, in-depth discussion of high-dimensional cytometry experimental design and data cleaning is beyond the scope of this review, but it has been thoroughly discussed elsewhere (see (22, 25–27) for several comprehensive descriptions).

In general, the best way to guard against spurious results and data misinterpretation is a fastidious approach to experimental design before starting an experiment as well as attention to data cleaning and quality checks after data have been collected. During the experimental design phase of a project, careful consideration should be given to the inclusion of batch controls, using consistent experimental protocols and reagents, and balancing case and control groups during sample acquisition. After data are obtained, it may need to undergo compensation to correct for spillover from overlapping fluorochromes or metal isotopes, transformation to the logarithmic or hyperbolic arcsine scale to improve data scaling, and normalization to correct for batch effects, technical variation between cytometers, and longitudinal differences in samples collected across broad timescales (i.e., weeks or months) (19, 28–30). Finally, data should also undergo a "sanity check" for quality assessment. This may include detecting outliers, technical errors in individual samples, or larger-than-expected between-sample variability and can frequently be accomplished by visualizing data distributions and computing summary statistics to discern how much each sample deviates from the norm (31, 32). Often, machine learning tools are applied to data that have not yet been manually explored to identify routine sources of error; instead, these sources of error are left to quietly wreak havoc on the analysis until the late stages of a project in which they are more difficult to identify (and after a great deal of one's time may have already been spent). Ensuring appropriate experimental design and quality control before undertaking any complex analyses can help to avoid these frustrating pitfalls.

Step 1: Dimensionality Reduction

Dimensionality reduction is a common method of succinctly visualizing single-cell data either to reveal broad trends in how cells are distributed in high-dimensional space or to roughly assess data quality across multiple experiments or data sets. Here, we discuss several methods of dimensionality reduction including principal component analysis (PCA), *t*-distributed stochastic neighborhood embedding (t-SNE or viSNE), and uniform manifold approximation and projection (UMAP). Example applications of each of these dimensionality reduction approaches to a recently published cancer cytometry data set are provided in Figure 3.

Principal component analysis

PCA is one of the most commonly used dimensionality reduction techniques and is often the first algorithm applied to new high-dimensional data sets. PCA reduces the dimensionality of an input data set by recombining its variables into so-called *principal components* (PCs), a set of new, uncorrelated variables that are rank-ordered by the amount of variance from the original data that they explain. Thus, PC1 explains the largest amount of variance from the input data set, PC2 explains the second largest amount, and so on. Importantly, each PC represents a linear combination of the original variables and, for cytometry data, can be conceptualized as a composite dimension of markers (or other cellular features) that contain similar information across all of the cells being analyzed (34).

In general, PCA is often used to visualize the structure of high-dimensional cytometry data by plotting cells along the first two (or sometimes three) PCs, and this can be useful for observing broad separations in cellular phenotype (7). In addition, most statistical software capable of performing PCA will also report the contribution of each individual marker to all of the PCs (called "loadings"). Factor loadings can be helpful in determining which markers contribute most of the variance to a data set and can be used in preliminary experiments to narrow down an antibody panel for high-dimensional flow cytometry or mass cytometry (for an example of this approach, see (35)).

Both because of its rapid compute time (due to its relationship with a linear algebra concept called the *singular value decomposition*, see Glossary) and lack of tuning parameters, PCA is an exceedingly convenient tool for analyzing broad patterns within a data set (34). For instance, the EuroFlow Consortium recently used PCA to develop a highly-sensitive method for detecting MRD in B-cell precursor acute lymphoblastic leukemia (BCP-ALL) (36, 37). In their study, bone marrow aspirates from 178 BCP-ALL patients were collected before treatment, after induction therapy, and 1 year after ending treatment and were analyzed using PCA in order to develop an antibody panel that best separated leukemic blasts from healthy B-cell precursors. Yet despite PCA's usefulness

for tasks like this, it also suffers from an inability to meaningfully represent highly complex, nonlinear relationships between variables. Biological data are inherently nonlinear because of the complex regulatory structures that abound in molecular biology—including processes like thresholding, saturation, signal amplification, and both positive and negative feedback. Thus, biological variables commonly have polynomial, exponential, or otherwise highly complex relationships with one another. This results in irregular distributions and relationships that may not be easily captured by PCA (38). Because of these common properties of biological data, dimensionality-reduction algorithms that can accommodate nonlinear relationships are often used to detect more subtle relationships than those represented by PCA.

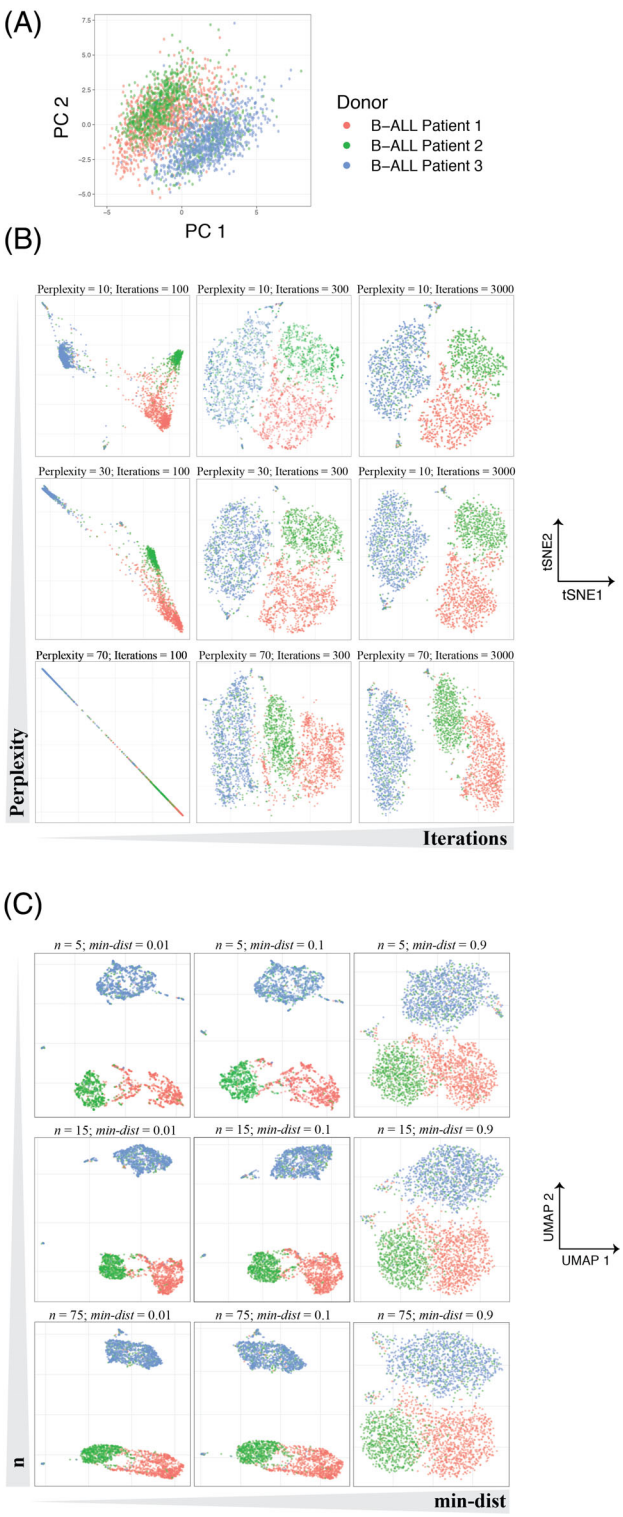

Fig 3. Dimensionality reduction using three commonly used approaches: PCA, t-SNE, and UMAP. A total of 10,000 cells were subsampled from three BCP-ALL patient samples analyzed using mass cytometry. Data were obtained from the GitHub repository from Good et al. (33). (**A**) Two-dimensional plot of three patient samples along their PC1 and PC2 axes. Note that PCA does not require the user to set any hyperparameters and will return the same result each time it is used. (**B**) Two-dimensional plot after performing t-SNE on the same cells as in (A) across several t-SNE hyperparameter values. Note that samples fail to separate from one another when the number of iterations is too low and that neither intersample distances nor dispersion are conserved across perplexity settings. (**C**) Two-dimensional plot computed using the same cells as in (A,B) across varying levels of *min-dist* and *n*. Slightly different embeddings result from different hyperparameter settings, although global relationships are more robust to these changes than those observed in t-SNE embeddings.

T-distributed stochastic neighbor embedding

Although PCA is limited to the detection of linear patterns, not all dimensionality reduction algorithms are. The first of the nonlinear algorithms we will discuss is t-SNE (often also called "viSNE" when used for visualization) (39). The t-SNE algorithm has been implemented in most programming languages commonly used for scientific computing including R, Python, and MATLAB. In addition, it is available for use as a graphical user interface (GUI) on both the FlowJo and Cytobank analysis platforms.

Like PCA, t-SNE analyses start with high-dimensional input data in which single cells are associated with corresponding measurements of protein marker expression in flow cytometry, sequence reads in single-cell nucleic acid sequencing, or cellular neighborhood information in multiplexed imaging. However, whereas PCA reduces dimensionality through a series of linear transformations based on the input data's *global* variance structure, t-SNE more accurately captures nonlinear relationships by emphasizing differences in the data's *local* structure (40). In brief, it does so in three steps. First, the pairwise distances between individual cells and each of their close neighbors are calculated in high-dimensional space and represented as a set of normal (Gaussian) probability distributions. Second, each cell is placed randomly on a pair of arbitrary axes to give an "initial" two-dimensional representation of the data in which distances between close neighbors are calculated using the t-distribution (41). And third, the probability distribution of the high-dimensional and initial 2-dimensional representations is compared, and the two-dimensional

representation is iteratively adjusted until it matches the high-dimensional representation as closely as possible. Ultimately, this means that t-SNE analysis places similar cells close to one another in the resulting plot such that distinct cellular subpopulations emerge visually (39).

Importantly, t-SNE requires the user to set several hyperparameters before an analysis is run—including the algorithm's number of iterations, learning rate, and perplexity—and these parameters can have significant effects on the final result. Perhaps the most important of these values is "perplexity," which represents the rough balance between the input data's local and global structure that is emphasized in the construction of t-SNE's low-dimensional representation. When perplexity is low, each data point is assumed to have a small number of close neighbors and local structure predominates; when perplexity is high, the opposite is true (41). Thus, a recommended best practice for conducting a t-SNE analysis is to test a variety of perplexity values (keep in mind that the recommended range is 5–50) on a given data set and observe which underlying patterns in the data are consistently observed (39). In addition, it is important to test multiple values for the number of iterations that the algorithm will use when constructing the low-dimensional representation, as values that are too low will fail to represent the data accurately. Importantly, several variations on the t-SNE algorithm—including hierarchical t-SNE (HSNE) (42) and opt-SNE (43)—have been developed since its initial description to improve its accuracy and compute time on large data sets.

t-SNE has been successfully applied to variety of high-dimensional cytometry data sets in the study of human cancer. For example, in t-SNE's initial application to cytometry data, Amir et al. used it to show that bone marrow aspirates taken from acute myeloid leukemia (AML) patients contain cells distinct from both healthy bone marrow populations and one another (39). More recently, t-SNE analysis has been used in multiple studies characterizing the differences between circulating and infiltrating immune cell populations in patients with solid tumors. For example, one recent study applied t-SNE analysis to mass cytometry data acquired from hepatocellular carcinoma (HCC) patient biopsies to show that immune cells within the HCC tumor microenvironment express higher levels of immunosuppressive surface markers than those outside the tumor. Specifically, the authors were first able to identify subsets of tissue resident memory CD8+ T-cells and T-regulatory cells (T-regs) expressing high levels of T-cell exhaustion markers—including PD-1, Tim-3, and Lag-3—after these cells segregated from other cell types on a series of t-SNE plots (44). Furthermore, a similar approach demonstrated a nearly identical result in a recent study of glioblastoma multiforme (GBM), in which t-SNE analysis helped to identify GBM tumor-resident T-regs cells that, when compared to T-regs in circulation, demonstrated both higher PD-1 expression and an elevated molecular signature of T-cell exhaustion (45). Together, these examples indicate that t-SNE can be especially useful during the exploratory phase of data analysis by parsing out broad population dynamics and standout cellular subsets on easily visualized, low-dimensional plots.

Yet, while t-SNE is a powerful tool for representing high-dimensional, nonlinear patterns in two or three dimensions, it can be easy to misinterpret. Even when t-SNE's tunable parameters are appropriately set, the algorithm does not preserve density or global distances between observations. This means two important things. First, it means that the relative size of a subpopulation on a t-SNE plot does not necessarily correspond to its actual size in high-dimensional space. Second, it means that the distance between distant "clusters" in a t-SNE plot does not always accurately reflect their similarity to one another (11, 41). Thus, interpreting distances on a t-SNE plot quantitatively can be misleading, although some studies have precariously done so to make claims regarding leukemic cell identity along the hematopoietic developmental trajectory (46, 47). Importantly, it should also be noted that t-SNE is stochastic, which means that it can yield slightly different low-dimensional representations when applied to the same data multiple times, although differences tend to be minor when the algorithm's parameters are chosen well (41).

Uniform manifold estimation and projection

While t-SNE is the most commonly used nonlinear dimensionality reduction technique in high-dimensional cytometry data analysis, it is limited to analyzing a relatively small number of cells due to its slow computation time (which scales quadratically with the number of cells being analyzed) (40). Due to this and other constraints on t-SNE's performance, an algorithm called UMAP was recently developed as an alternative dimensionality reduction approach (48, 49). Because UMAP was so recently developed, there are limited examples of its application to the study of cancer biology specifically; however, UMAP is becoming increasingly widespread in the field of bioinformatics in general, which is why it warrants discussion here (50–52). UMAP has been implemented as software packages in both Python and R.

Like t-SNE, UMAP seeks to represent the high-dimensional structure of an input data matrix in low-dimensional space such that local relationships between nearby cells are conserved. Unlike t-SNE, however, UMAP leverages manifold theory and Riemannian geometry to accomplish this by first approximating the high-dimensional surface on which the data sits, then utilizing a weighted k-nearest neighbor graph architecture to project that surface onto a low-dimensional layout (48). While UMAP is derived using mathematical theory that most biologists will be unfamiliar with, in practice UMAP and t-SNE can be used for similar purposes. UMAP's particular strengths derive from its significantly faster compute time and greater emphasis on global data structure relative to t-SNE, as well as its ability to add new observations to an existing plot, of which t-SNE is not directly capable (49). This ability to embed additional data points on an existing plot is particularly useful for biological data analysis, as it has direct applications in longitudinal disease monitoring (i.e., tracking individual patients over time throughout disease progression or treatment) as well as in the detection of batch effects when new samples are analyzed. Importantly, UMAP's greater preservation of a data set's global structure means that comparing distances between clusters on a

UMAP embedding might be a bit more meaningful than doing so on a t-SNE plot (53). However, because local distances are used to compute both t-SNE and UMAP embeddings, global distances (particularly between very distant data points) are difficult to interpret. In most cases, it is less precarious to simply compute distances in the original, high-dimensional data space, making use of the distance metric that best suits your particular data set (see "Distance metrics" in Glossary).

UMAP requires the user to tune several hyperparameters, including *n*, the number of neighbors that UMAP will use to learn local data structure; *d*, the target number of output embedding dimensions; *min-dist*, the minimum distance allowed between close points in the low-dimensional representation; and *n-epochs*, the number of iterations the algorithm should use to find a stable low-dimensional representation (48). Arguably, the most important of these parameters is *n*, for which larger values will emphasize global data structure over local structure (similar to t-SNE's perplexity parameter; see Fig. 3c). By contrast, *min-dist* is a purely aesthetic parameter for which low values will result in more closely packed plots. In general, users should expect to test a variety of values for *n* and *min-dist* while using values of *d* and *n-epochs* that provide stable output in the desired number of dimensions (generally two or three).

When choosing between PCA, t-SNE, and UMAP in practice, there are several considerations to keep in mind. For instance, PCA is an optimal choice when computational speed, interpretability, and simplicity (due to PCA's lack of hyperparameters) are important. By contrast, t-SNE and UMAP are stronger choices when an analysis requires the visual separation of cell types whose measurements have highly nonlinear relationships with one another or whose differences are poorly resolved using PCA in two or three dimensions. Finally, longitudinal data analyses are best performed using PCA and UMAP due to their ability to embed new samples into a coordinate system that has already been computed on previous samples, thereby allowing the direct comparison of old and new data. That being said, outside these specific criteria, many data scientists also choose between these dimensionality reduction methods using trial-and-error—generally by starting with PCA due to its speed and working up to slower, more complex methods like UMAP and t-SNE if necessary.

Step 2: Clustering

While dimensionality reduction algorithms are useful for exploratory data analysis and visualization, they do not explicitly compare cell subpopulation structure between samples. To accomplish this, investigators can utilize clustering algorithms that stratify cells into quantifiable subsets that are (for the purposes of the clustering) assumed to be similar. This allows the bulk characteristics of different clusters to be compared both to one another and across sample types. While there are many kinds of clustering algorithms that work in different ways, they all seek the explicit goal of assigning observations into distinct groups such that similar observations are assigned to the same group and dissimilar observations are assigned to different groups. While clustering ultimately leads to the loss of single-cell resolution when applied to cytometry data, it also allows investigators to make inferences about which phenotypes—shared among many cells—are present in their data. Such inferences are frequently important in cancer biology, where deconvolving heterogeneous mixtures of cells within bulk tumor populations is a common experimental goal (54, 55).

In this section, we give an overview of the most common clustering approaches used in the analysis of cancer cytometry data including hierarchical clustering, k-means clustering, density-based clustering, and graph-based clustering (Fig. 4). We also provide examples of how these algorithms have been applied to specific questions in cancer biology and adapted specifically for the analysis of cytometry data. Importantly, many existing clustering tools are associated with built-in visualization strategies, but clusters can also be visualized using any of the dimensionality reduction strategies described in the previous section.

Hierarchical clustering

By far the most commonly used clustering technique in bioinformatics is hierarchical clustering. Hierarchical clustering operates either by iteratively combining observations into progressively larger groups (the bottom-up or "agglomerative" approach) or by iteratively dividing observations into progressively smaller groups (the top-down or "divisive" approach) (56). In the agglomerative approach, each cell is initially assigned to its own cluster and, as the algorithm iterates through multiple steps, clusters are combined with similar clusters one-by-one until the desired number of clusters is reached. In the divisive approach, all cells are initially assigned to a single cluster that is repeatedly divided into smaller clusters such that the dissimilarity between the resulting clusters is maximized. As in the agglomerative approach, this splitting process continues until the desired number of clusters is reached (57). In both approaches, the investigator must choose the desired number of clusters in the final result a priori—and while this is the only hyperparameter that hierarchical clustering requires, users should be cautious to avoid choosing a number of clusters that is too small, which can limit the resolution of analysis by forcing dissimilar cell types to be grouped together spuriously.

Importantly, hierarchical clustering approaches vary in how they define the dissimilarity or distance between clusters. In *single-linkage hierarchical clustering*, the distance between two clusters is defined as the smallest distance between any point in the first cluster and any point in the second cluster. By contrast, *complete-linkage hierarchical clustering* does so by instead using the largest distance between any data point in the first cluster and any data point in the second cluster. As something of an intermediary between single-linkage and complete-linkage, *average-linkage hierarchical clustering* uses the average distance between all data points in one cluster and all data points in a second cluster to define inter-cluster distance (58). Each of these linkage types produce slightly different results, and they are each capable of using a variety of

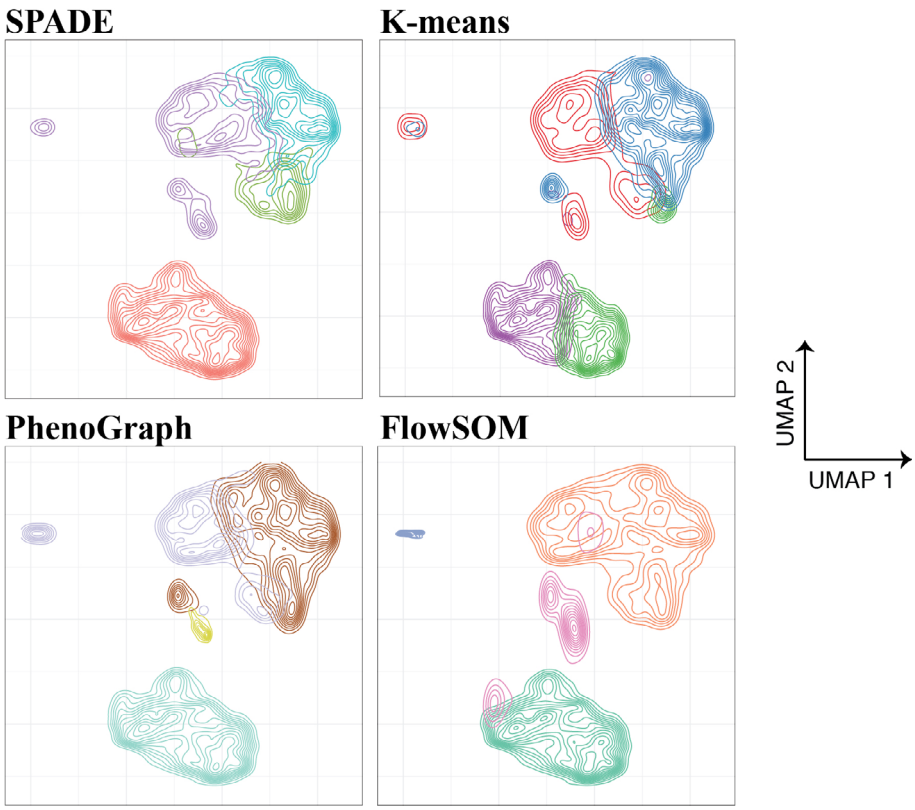

Fig 4. Comparison of clustering results using SPADE, K-means clustering, PhenoGraph, and FlowSOM. A total of 10,000 cells were subsampled from each of three BCP-ALL patient samples analyzed using mass cytometry. Data were obtained from the GitHub repository from Good et al. (33) and analyzed using R implementations of SPADE, k-means clustering, PhenoGraph, and FlowSOM. PhenoGraph automatically detected the presence of four clusters, so this number of clusters was specified for the three remaining algorithms in order to compare results; otherwise, default parameters were used. Contour plots were embedded within UMAP axes computed using all 30,000 subsampled cells, with distinct clusters identified by each algorithm represented with a unique color in each panel. Across all clustering methods, markers used for clustering were the following: CD19, CD20, CD24, CD34, CD38, CD127, CD179a, CD179b, IgM (intracellular and extracellular), and terminal deoxynucleotidyl transferase. Notably, different clustering approaches identify subtly different cellular subsets even within this relatively simple data set. Often, iteratively testing different clustering approaches, visualizing the results, and adjusting hyperparameters can help to determine which method fits best for one's particular data set.

distance metrics. While Euclidian distance is a common choice, other similarity metrics (such as Manhattan distance, Pearson's correlation, and others) are frequent alternatives.

In the analysis of high-dimensional cytometry data in particular, a commonly used application of hierarchical clustering is spanning-tree analysis of density-normalized events (SPADE), an algorithm best known for its application to the analysis and visualization of mass cytometry data, originally applied by Bendall et al. (3) After an initial density-dependent subsampling step, SPADE utilizes agglomerative, single-linkage hierarchical clustering in which the distance between observations is calculated using the absolute value (L1) norm (59). After identifying a user-specified number of clusters, SPADE constructs a minimum spanning tree (MST) diagram that connects similar clusters to one another and embeds them in a two-dimensional plot. In SPADE plots, clusters are represented as nodes within the MST whose size corresponds to the number of cells in each cluster, and whose color can be used to represent marker expression. As a widely used tool, SPADE has been implemented in R, C++, and Java and is also available as a GUI within Cytobank. Recently, a deterministic implementation of SPADE was developed such that its down-sampling and MST construction steps are no longer stochastic; however, this version has not yet received widespread use in the study of cancer (60).

While SPADE was initially developed to cluster and visualize subsets of hematopoietic lineage cells in the healthy immune system, it has since been applied to single-cell cancer data to help parse bulk tumor heterogeneity. In one recent study, investigators applied SPADE to data acquired from blood samples collected from nine patients with secondary AML. After clustering, the authors were able to identify several populations of $CD34^{+}CD38^{+}$ AML cells whose intracellular signaling program (specifically, phosphorylated STAT3 and STAT5 expression) responded to thrombopoietin stimulation differently than healthy controls, revealing a potentially targetable population for therapeutic development (47). SPADE has also recently been used to identify a subset of

metabolically impaired, tumor-infiltrating CD8$^+$ T lymphocytes in clear cell renal cell carcinoma (ccRCC) patient samples analyzed using mass cytometry. In this study, analysis of SPADE plots revealed that a subset of ccRCC CD8$^+$ T-cells exhibited a less activated surface phenotype compared to circulating T-cells in ccRCC patients' peripheral blood, with functional experiments confirming that this resulted from ccRCC T-cells having small, fragmented mitochondria, and glucose metabolism deficiency (61).

Another algorithm with similar visualization capabilities to SPADE is FlowSOM, a recently developed clustering approach that also returns its result in the form of an MST. Unlike SPADE, FlowSOM does not require downsampling of its input data, which both significantly reduces its computation time and allows it to analyze a larger number of cells at once (62). This is due to the fact that, instead of directly performing hierarchical clustering, FlowSOM first assigns single cells to clusters using an artificial NN called a *self-organizing map* (SOM). SOMs are trained by assigning cells to their most similar node in a discretized grid of the input space, then iteratively adjusting the position of all nodes until similar cells are grouped together in an a priori specified number of clusters (62). After clusters are identified, they can then be visualized as an MST and combined into larger meta-clusters in a final agglomerative hierarchical clustering step. While FlowSOM has been applied to studying cancer less than to studying the immune system, one recent study utilized FlowSOM to analyze melanoma cells biopsied from patients both immediately before initiating chemotherapy and after 4 weeks of treatment. The investigators' analysis revealed one cluster of melanoma cells with a consistent surface marker profile that persisted throughout treatment, suggesting a potential role for this rare subtype in treatment-resistant melanoma (63).

K-means clustering

Another commonly used clustering algorithm is K-means clustering. Like hierarchical clustering, the K-means algorithm requires a user to explicitly specify the numbers of clusters (called K) to be identified within the input data (64). Once K is chosen, the algorithm performs a three-step procedure. First, it randomly assigns each observation to one of the K clusters. Second, it computes the mean of each input variable across all data points within each cluster. Third, it reassigns each observation to the cluster whose mean is closest to it, regardless of which cluster that observation was assigned to previously. Thus, the second and third steps of this algorithm can be repeated many times until the means of the K clusters change very little (or not at all), indicating that all observations have been given a stable cluster assignment.

K-means clustering can be used similarly to any of the hierarchical clustering approaches described above, so we avoid going into further detail here. While k-means clustering is commonly used in bioinformatics overall, its tendency to produce spherical clusters has been shown to impair its performance on high-dimensional cytometry data sets, which often

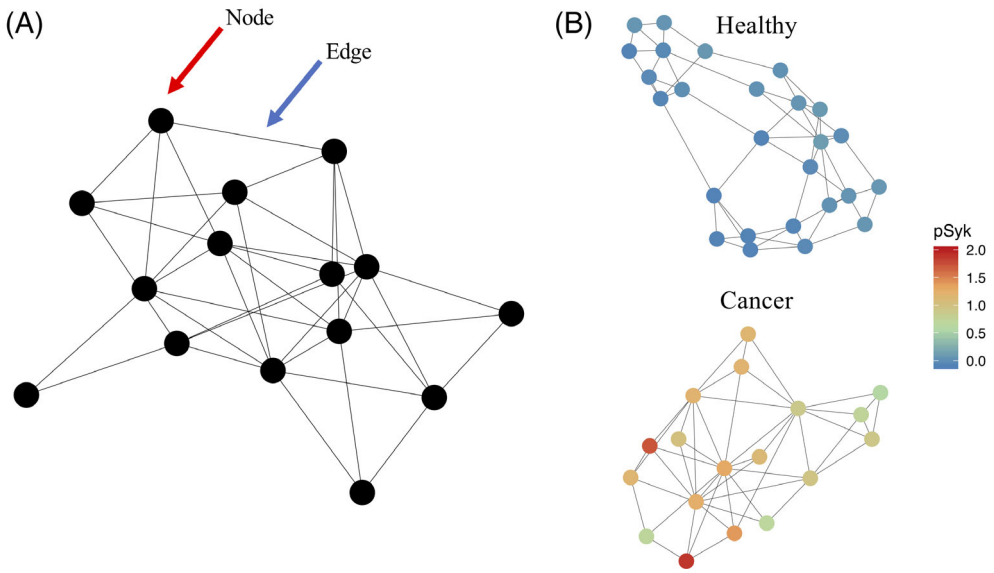

Fig 5. Graph architectures can be used to represent cytometry data. (**A**) Schematization of a "graph," a data structure that expresses observations as *nodes* and the relationships between observations as *edges*. The red arrow points to a node; the blue arrow points to an edge. (**B**) Example graphs constructed from cytometry data collected via CyTOF (data taken from Good et al. (33)). This is an example of a graph representing single-cell cytometry data: In it, the nodes represent clusters of single-cell observations and the edges represent relationships between those nodes. In this case, a k-nearest-neighbor graph was built, meaning that each cluster is connected to the k clusters to which it is most similar (using Euclidean distance and k = 3). (B) Clustering was performed by applying PhenoGraph in healthy and leukemic samples. Each cluster's expression level of phosphorylated Syk protein (pSyk), a relapse-predictive feature in pediatric BCP-ALL, is indicated colorimetrically for each node. This example graph illustrates how biological parameters can be depicted by using a graph-based representation.

contain irregularly shaped cell subpopulations (65). For that reason, variations on k-means clustering such as flowMeans, an algorithm that automatically combines similar clusters detected by k-means clustering, are more commonly used (66). Perhaps most importantly, however (and often a source of confusion among novice data scientists), we point out that K-means clustering is different than K-nearest-neighbor classification (and other K-nearest-neighbor algorithms) despite their names sounding similar. Rather, K-nearest-neighbor algorithms—some of which are described below—revolve around identifying a user-specified number of data points (also called *K*) with the smallest distances to a particular observation and then performing some computation using those data points. While K-nearest neighbor algorithms are useful for a variety of applications, K-means clustering does not involve calculating any point's K nearest-neighbors.

Graph-based clustering

In addition to hierarchical and k-means clustering, an increasingly common approach to cell subset detection in high-dimensional cytometry is utilizing graph-based community detection algorithms to identify distinct subpopulations of cells. In graph-based clustering, individual cells are represented as *nodes* or *vertices* that are connected to one another along a set of links (called *edges*) based on their phenotypic similarity to one another. Together, this set of nodes and edges is referred to as a "graph," an abstract data structure that emphasizes the relationships between data points (and can be visualized two-dimensionally, as in Fig. 5). There are many ways to represent a cytometry data set in graph form, and depending on the kind of graph being constructed, individual cells may have many or few connections to other cells. Regardless of the specific kind of graph used, however, graph representations allow for the application of powerful computational tools to single-cell data in ways that have yielded many unique discoveries.

An important graph-based clustering algorithm in the study of cancer cells is Louvain clustering, an approach that was first developed to detect interconnected communities in mobile phone networks (67). The Louvain algorithm works by repeatedly applying two computational steps. In the first step, the algorithm assigns each node to its own cluster (similar to agglomerative hierarchical clustering). Then, the algorithm combines clusters iteratively until the graph's overall "modularity"—a measure of the connection density within clusters relative to the connection density between clusters—is locally maximized (68). In the second phase, the algorithm constructs a higher order graph by creating supernodes out of clusters identified in the first phase. These two phases are repeated until the graph's modularity can no longer be increased, at which point the final clustering result is returned. Louvain clustering has been implemented in a variety of programming languages and is generally considered both relatively fast and stable relative to many other graph-based algorithms (69).

In the analysis of cytometry data, one variation of Louvain clustering was used to develop the PhenoGraph algorithm, a clustering method specifically developed to study cancer cell heterogeneity (35). In brief, PhenoGraph works by representing the input data set as a graph in which each cell is connected to each of its k nearest neighbors using Euclidian distance (where k is a user-specified hyperparameter). Using this initial representation, PhenoGraph then builds a second graph in which the similarity between cells is redefined according to their number of shared neighbors using the Jaccard coefficient (70). Louvain clustering is then performed on this graph, yielding a clustering result that can detect cell subpopulations as rare as 1/2,000 and that robustly reproduces manual gating (35). Importantly, PhenoGraph accepts only a single user-defined parameter: k, the number of nearest neighbors used to construct the initial graph, and automatically detects the optimal number of clusters present in the data set without a priori specification.

When it was first presented, PhenoGraph was applied to cytometry data collected from pediatric AML patient bone marrow aspirates. While other clustering methods failed to partition the data into interpretable subpopulations, PhenoGraph clustering revealed that AML cells were distinguishable mainly by their signaling phenotypes (particularly with regard to phosphorylated STAT5 and phosphorylated AKT) despite their high variability in surface phenotype. Since this initial study, PhenoGraph has also been used to characterize rare, progression free survival-associated macrophage and lymphocyte populations in human renal cell carcinoma samples (71), to survey the differences in immune composition between circulating and tumor-infiltrating immune cells in lung adenocarcinoma (72), and to identify a novel neutrophil-specific progenitor cell type in human bone marrow that confers proleukemic activity when transplanted into immunodeficient mice (73). In each of these cases, PhenoGraph was used to identify both small clusters with high-resolution as well as larger "metaclusters" of common cell types shared between tissues and patients.

Density-based clustering

The final clustering approach that we will discuss is density-based clustering, which partitions high-dimensional data into discrete subpopulations based on local densities. Density-based algorithms are capable of identifying clusters without assuming a particular cluster size or number and are well-suited for cytometry experiments in which a large number of observations are collected. There are many ways to calculate density estimates in single-cell data, with particular methods performing best for specific applications over others (74).

One robust density-based algorithm developed specifically for analyzing high-dimensional cytometry data is X-shift (75). Using k-nearest-neighbor density estimation, X-shift computes a local density estimate for each cell in the data set. It then searches for local density maxima among these estimates and labels each of these maxima as a cluster centroid. All remaining data points are then assigned to the nearest cluster along a density-ascending path. After clusters are identified in this way, the final step of the algorithm checks for

the presence of local density minima between neighboring centroids—if no clear minima are found, neighboring clusters are merged. Clusters are also merged if they are separated by a Mahalanobis distance of less than 2.0, a value based on the theoretical density-separation cutoff of the normal distribution (64). Importantly, the optimal value for the parameter k (signifying the number of nearest neighbors used in local density estimation) is automatically selected by the X-shift algorithm, resulting in an autonomously identified number of clusters. The optimal value of k is selected using line-plus-exponent regression—in other words, k is located at the "switch point" where the number of X-shift clusters begins to increase exponentially. Thus, the switch point is used to set the value of k in order to avoid over fragmenting the input data into an exceedingly larger number of clusters.

Despite the high degree of computational time required to conduct density-based clustering (due to the inherent slowness of exhaustively computing local densities), X-shift's performance has been validated extensively on flow cytometry data sets of healthy immune cells (9). More recently, X-shift has also been applied to cancer immunology. In one recent study, X-shift clustering was used to identify distinct populations of Reed-Sternberg (RS) cells in classical Hodgkin Lymphoma (cHL) patient biopsies (76). In addition to demonstrating that some populations of RS cells lose MHC Class I expression relative to control lymphoid tissue, X-shift allowed for the identification of multiple differences in cells of the cHL tumor microenvironment, including an expansion of immunosuppressive T helper 1 cells and T-regs. X-shift has also been applied to the dissection of solid tumor cell heterogeneity by identifying a relapse-associated tumor cell subset particularly high in vimentin, HE4, and cMyc expression in 17 high-grade serous ovarian tumor samples (77).

Together, the clustering methods described above constitute a broad range of strategies for cell subset detection in cancer cytometry data. It is important to note that each of these clustering algorithms is most useful with specific analytical goals in mind. For instance, when fast compute times are important, k-means clustering (and its variants) as well as flowSOM are generally optimal, whereas PhenoGraph, X-shift, and SPADE are significantly slower. By contrast, hierarchical clustering is a useful choice when multiple sets of nested, nonmutually exclusive clusters are desired, and either PhenoGraph or X-shift can be helpful when you are unsure how many clusters are present in your data set (due to their ability to automatically detect cluster number). Furthermore, PhenoGraph is particularly useful for detecting especially rare cell subtypes despite requiring a fair amount of computational time and memory resources. And when understanding the phenotypic relationships between clusters is important, both SPADE and flowSOM's built-in minimum-spanning tree visualization capabilities can make them strong candidates for your analysis. Importantly, we also note that implementation details differ between each of these methods—only flowSOM, SPADE, and X-shift have been implemented with GUIs, so groups unfamiliar with R or Python may find it more difficult to use other tools.

Step 3: Correlative Biology and Predictive Modeling

Using a combination of dimensionality reduction, visualization, and clustering tools, it is possible to identify subpopulations of cells with distinct molecular characteristics. But once these clusters are identified, how can we tell which ones (if any) are biologically or clinically important? To answer this question, we can look for associations between cluster characteristics—such as relative abundance, surface marker expression, and intracellular signaling program—and the clinical or experimental conditions of interest within the study. Using statistical techniques, it is possible to identify cell subsets or biological features correlated with (or predictive of) clinical or experimental outcomes. Once these cell populations or features have been identified, they can be further characterized and visualized in turn using the dimensionality reduction and clustering methodologies discussed in previous sections.

While many predictive modeling approaches have been adapted specifically for high-dimensional cytometry data, the majority of them are based on highly vetted analytical methods originally developed for transcriptome analysis (78). We discuss some key algorithmic tools including Citrus (cluster identification, characterization, and regression), Statistical Scaffold, and Cox survival analysis below.

Differential Abundance and Differential Expression Analysis

Most cytometry experiments include samples from two or more categories that investigators hypothesize will be different from each other in a meaningful way. In cancer biology, for instance, an experiment might include bone marrow aspirates collected from both leukemia patients and healthy donors, and investigators might expect to see differentially activated intracellular signaling programs between these two sample types. Alternatively, an experiment might analyze diagnostic samples versus relapse samples, primary tumors versus metastases, samples taken before treatment versus after treatment, or biopsies collected from patients with good prognoses versus poor prognoses. In each of these cases, an important experimental goal is identifying which characteristics represent the most important differences between sample types.

Citrus (cluster identification, characterization, and regression) is an algorithm that was designed specifically to tackle this question (79). In its first step, Citrus uses agglomerative hierarchical clustering to group phenotypically similar cells based on marker similarity (using Euclidian distance). For each cluster, Citrus then calculates several features on a per-sample basis—including the proportion of cells in each cluster per sample and the median expression level of each marker within each cluster. These features are used to construct an $M \times N$ matrix that represents the N features computed by Citrus across each of the M samples being analyzed. Importantly, Citrus records all clusters identified in the entire clustering hierarchy and uses them for downstream analyses—this means that each cell will be assigned to multiple nested clusters that are all represented in the Citrus feature matrix.

Using this feature matrix, Citrus then trains a supervised model to predict which sample group a sample belongs to based on its cluster features. Specifically, Citrus uses either lasso-regularized multinomial logistic regression or nearest shrunken centroid classification for the predictive step. Multinomial logistic regression is a type of generalized linear model in which input features (in this case, our cluster features) are used to predict the relative probability that a sample belongs to a particular class (80). By contrast, nearest shrunken centroid classification simply assigns each observation to the sample category whose noise-adjusted centroid is closest (81). These models produce comparable results and are fitted across a range of "regularization thresholds" that determine how many (or how few) features are included in the final model. Both types of model are validated using 10-fold cross-validation, and the optimal model is ultimately selected based on maximum accuracy. Citrus also provides some built-in visualization of its results, all of which are easily accessible in its Cytobank and R implementations.

By combining clustering, generalized linear modeling, and regularization, Citrus automatically identifies which cytometric features are most closely associated with a particular experimental or clinical outcome. Thus, Citrus performs what is often referred to as *feature selection*—a process whereby an algorithm identifies a data set's most informative variables out of a large group (17). Because it considers both cluster size and marker expression, Citrus can also be described as performing *differential abundance analysis* (comparing cluster size between conditions) and *differential expression analysis* (comparing clusters' marker expression levels between conditions), both of which are standard bioinformatic tools. Importantly, Citrus requires at least eight samples for each experimental group being compared in order to maintain sufficient statistical power to differentiate between true differences and normal inter-sample variability (noise) (80). This limitation is important to consider when designing experiments—if one's sample number is limited, Citrus may not provide reproducible results.

Another algorithm that takes a similar approach is Scaffold (single-cell analysis by fixed force- and landmark-directed maps) (82). Initially developed purely as a visualization tool, Scaffold's main functionality is to organize cell clusters on a force-directed graph based on their relative similarity to manually gated reference cell populations (83). However, the algorithm was later updated to include the option to apply the significance analysis of microarrays permutation-based significance test to the clusters being plotted (84). To do so, the updated "Statistical Scaffold" algorithm first calculates the same feature matrix as Citrus. Then, it randomly permutes the sample labels to estimate each feature's expected difference between sample types solely due to chance. Using this approach, a false-discovery rate for the set of features can be computed, and statistically significant differences can be assessed (83). As a differential abundance and expression analysis tool with useful visualization capabilities, Scaffold has been applied to the study of chimeric antigen receptor (CAR) T-cell dynamics in preclinical studies of novel CART cell therapies (85). However, Scaffold has seen limited use in studying cells outside the hematopoietic lineage, perhaps due to the often drastic difference between cancer cell phenotypes and those of their native, healthy lineages. Furthermore, due to Scaffold and Citrus's identical feature matrix calculations, they share similar sample size requirements. In order to maintain sufficient statistical power to detect relevant differences, we recommend a sample size of at least eight samples per group when using Scaffold as well.

Cox survival analysis

In addition to the analyses mentioned above, many translational studies include the explicit goal of predicting clinical outcomes directly from single-cell data. This is especially true in the study of cancer, in which predicting patient mortality, risk of recurrence, and other adverse outcomes are imminent areas of clinical need. Thus, single-cell studies of clinically annotated human samples may seek to quantify the relationship between cellular features and outcome variables such as patient survival or time to relapse. Importantly, understanding such relationships is becoming increasingly possible due to the emergence of large repositories of clinically annotated data sets such as the National Institute of Health's Cancer Genome Atlas and Therapeutically Applicable Research to Generate Effective Treatment (TARGET) program.

Statistical modeling for time-to-event outcomes such as mortality or relapse is typically accomplished using Cox regression, a supervised learning tool most commonly used in epidemiology (86). Like logistic regression (discussed above), Cox regression is a kind of generalized linear model in which input features are used to predict the "hazard rate" of a particular event happening over time (87). Once trained, a Cox model is capable of predicting the relative survival (or event-free survival) probability of a patient with a particular set of input features at any point in time. As generalized linear models, Cox models can be regularized similarly to logistic models, which allows for feature selection of only the most informative predictors. Thus, Cox regression can be a powerful and versatile tool in outcomes prediction when applied to the appropriate data set.

Conveniently, the Citrus algorithm (discussed at length above) is capable of fitting Cox models to perform survival analyses of multiparameter cytometry data (79). Using the same feature matrix computed for logistic regression, Citrus can be specified to train a lasso-regularized Cox model instead of a logistic model (80). Survival modeling in Citrus is performed using the same procedures as logistic classification, using k-fold cross-validation to build a model using the subset of cluster features most predictive of patient risk. Selected features are reported by Citrus and can be inspected for biological interpretation.

Although Citrus provides a convenient implementation of Cox regression, it has been used infrequently in the analysis of cancer data. This may be the case due to the Citrus workflow's inability to incorporate other relevant clinical variables—such as patient age, treatment regimen, or tumor

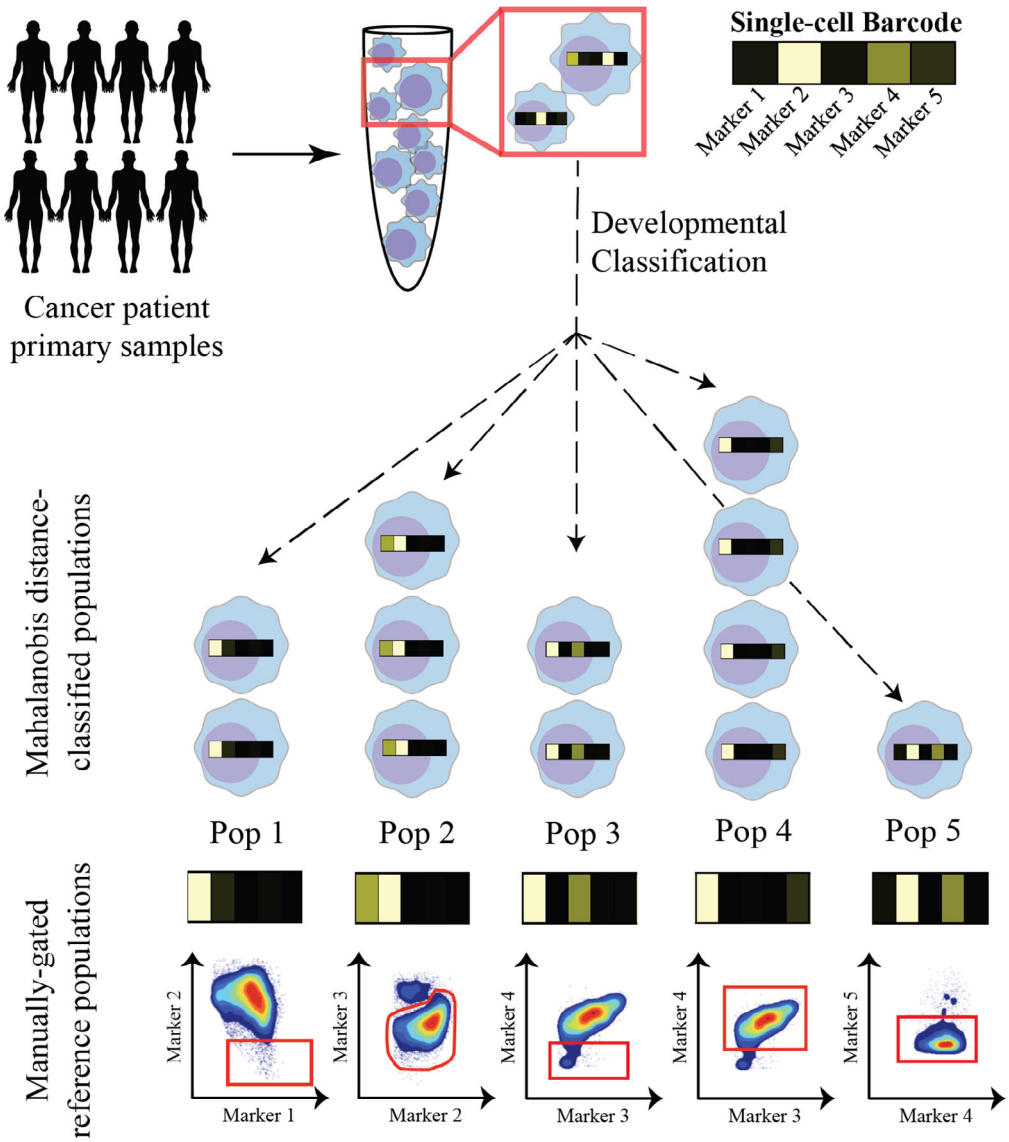

Fig 6. Schematization *of Good et al.'s single-cell developmental classifier. Using this approach, cancer cells are classified into their most analogous healthy cell type in normal lineage development in a series of two steps. First, healthy populations across lineage development are manually gated, and a single-cell "barcode" of marker expression values is computed for each manually gated subpopulation. Second, cancer cells are aligned with their most similar healthy subpopulation based on the Mahalanobis distance between their marker expression profile (or "barcode") and that of each manually gated population. Using this method, cancer cells can be classified into readily interpretable, "healthy-like" cell subtypes that each has unique properties.*

genetic subtype—into its survival analyses (see (22)). However, the general approach of applying regularized Cox models to cancer cytometry data is easily implemented in a wide variety of available R packages (notably *glmnet* (88), *edgeR* (89), *caret* (90), and others).

This strategy was recently used to combine the use of mass cytometry and clinical metadata to predict disease recurrence in a cohort of 60 patients with pediatric BCP-ALL (91). In this study, investigators first developed a supervised clustering algorithm (termed a "single-cell developmental classifier") by using Mahalanobis distance to align leukemic cells with their most similar developmental subpopulation along healthy hematopoietic development (Fig. 6). For each healthy-like cluster of leukemic cells, a feature matrix similar to Citrus's was calculated and used to train an elastic net regularized Cox model to predict patients' relapse risk with >90% accuracy (33). Based on the model's selected features, investigators were able to identify activated signaling in the mTOR and pre-BCR pathway in specific subpopulations of leukemic cells, thereby identifying potential therapeutic targets for further study in BCP-ALL.

In sum, each of the tools described above can be used to detect associations between single-cell measurements and

clinical or experimental outcomes in annotated samples. Citrus is a convenient tool for a variety of analyses including differential expression analysis, differential abundance analysis, and survival analysis. However, it is limited by both its sample size requirements as well as its inability to model variables outside its specific feature matrix (such as patient age or gender). Statistical Scaffold is similar to Citrus, but it places a larger emphasis on visualization and thus can be a more useful tool for creating plots to represent your data. Lastly, directly implementing cox, logistic, or any other kind of generalized linear regression can be an incredibly powerful and flexible tool, particularly because regularization approaches such as elastic net can make feature selection relatively easy. However, this approach is less straightforward than applying "off-the-shelf" algorithms like Citrus or Statistical Scaffold and may therefore require a larger amount of data preprocessing and feature extraction, such as Good et al.'s extraction of developmental lineage features from their data set before model building. Fortunately, there are numerous open-source tools for most data manipulation tasks, including the "Tidyverse" suite of data analysis tools in R and many other packages for data wrangling, visualization, and modeling (92, 93).

Conclusions and Topics for Future Consideration

We have presented a conceptual framework for understanding some of the most commonly used applications of machine learning to cancer cytometry data. Through a combination of dimensionality reduction, visualization, unsupervised clustering, and predictive modeling, it is possible to identify trends in cytometry data that point toward a novel understanding of basic and clinical cancer biology. By allowing high-dimensional data to be visualized in a smaller, more human-readable number of dimensions, dimensionality reduction provides a tool for exploring broad trends in one's data, detecting batch effects or other technical artifacts, and comparing samples longitudinally across disease progression or treatment. Likewise, clustering analyses allow for the detection of cell subpopulations in a data set through a variety of different approaches that make unique assumptions about how subpopulations are shaped, sized, and distributed. Finally, differential expression analysis, differential abundance analysis, and predictive modeling techniques can associate the biological measurements in a cytometry data set with other variables of interest, such as a patient's risk of relapse or their likelihood to respond to a particular treatment regimen. Thus, the specific goals and capabilities of each of these analytic steps can be used to guide an analysis from its early, exploratory stages to its targeted identification of a clinically or translationally important finding. Despite this, many areas for further development remain.

There are dozens of clustering algorithms that have been applied to cytometry data beyond those discussed here. Choosing a clustering approach for one's own study can be difficult, and the best advice is to try a few different approaches on simple data sets you already understand such as a control data set. This will allow you to assess the results based on your preexisting knowledge. Then, you can either increase the complexity of your input data gradually or go straight to your experimental data. Despite the continued development of increasingly sophisticated clustering approaches, an interesting recent trend is the emergence of clustering strategies that explicitly emphasize the interpretability of cluster identity. One approach is the developmental classifier developed by Good et al., which partitions leukemic cells into groups based on prior knowledge about healthy hematopoietic lineage development (91). Another approach is that of Marker-Enrichment Modeling (MEM), an algorithm that assigns text labels to preidentified clusters of cells based on their marker expression profiles relative to user-specified reference populations (94). MEM labels are meant to recapitulate historical naming conventions in cytometry in order to emphasize easy interpretation of cluster identity, regardless of the particular clustering method that was used to perform the analysis. Finally, two recently published algorithms—"GateFinder" and Hypergate—operate in a similar vein by automatically developing multistep, two-dimensional gating strategies for novel cell populations of interest within a data set (95, 96). Thus, these methods can be used to develop a sorting strategy for cell subsets identified using unsupervised clustering for follow-up experiments and additional characterization.

Regardless of how clusters are identified or annotated, one outstanding limitation of current analysis strategies is their loss of single-cell resolution (due either to clustering or to manual gating) before predictive modeling is performed. Because cells exist in a complex network of biological interaction and functional interdependence, this loss of single-cell information is likely to obscure important relationships between interacting cell types. This is especially true in the study of cancer, where understanding the interplay between individual tumor cells, the stromal microenvironment, and the immune system is generally understood to be of paramount significance. Although several recently developed algorithms are capable of predicting clinical outcomes from cytometry data without the use of clustering, methods that explicitly model potential interactions between cell types are yet to be developed (97, 98).

By contrast, current methods that *do* preserve the single-cell resolution of high-dimensional cytometry data sets are primarily focused on replicating manual gating procedures rather than on modeling clinical or biological outcomes. Such methods—including OpenCyto (99), flowClust (100), flowDensity (101), FlowLearn (102), and DeepCyTOF (103)—were developed to reduce the time and resources required for identifying "standard" cell populations as well as to reduce subjectivity in the gating process. Perhaps because cancer cell phenotypes are often much more heterogeneous than those of the immune populations on which these methods were developed, automated gating algorithms have not yet been applied extensively to cancer cytometry data. However, there is growing clinical interest in using automated gating strategies to detect MRD in some subtypes of acute leukemia in humans, which may represent an area of further development of these approaches as they pertain to the study of cancer (104).

Finally, with the progressive availability of large, publicly available multi-omics single-cell data sets, an exciting area for the future of cancer cytometry analysis is the application of deep learning. Because many current studies are often limited by the sparse availability of clinically annotated patient samples, deep learning is likely to overfit the small number of observations in most independent experiments. However, studies have begun to apply deep learning at the single-cell level—an approach that has been particularly effective in the application of convolutional NNs to computer vision problems in image cytometry (14, 105) as well as to batch correction, denoising, and clustering procedures on at least one large, multi-patient data set (106).

Together, the continued design, refinement, and the widespread use of machine learning algorithms in cytometry data analysis hold a great of promise in aiding future developments in our understanding and treatment of cancer. With the application of increasingly powerful and intuitive tools to larger and more diverse data sets, the field of cytometry is likely to provide important biological and clinical insights for decades to come.

Acknowledgments

We thank Dr. Nima Aghaeepour (Stanford University School of Medicine) and Dr. Brice Gaudilliere (Stanford University) for useful discussions. This work was supported by the US National Institutes of Health (National Cancer Institute 1F31CA239365-01 to T.J.K.). Kara Davis's work on this publication was funded by The V Foundation, the Andrew McDonough B+ Foundation, the Hyundai Scholar Hope Grant, and the Maternal and Child Health Research Institute. Pablo Domizi and Yu-Chen Lo's work was funded by the Leukemia and Lymphoma Society. Garry Nolan's work on the publication was funded by NIH: U54CA209971, U54HG010426, U19AI100627, and the Parker Institute for Cancer Immunotherapy.

Conflict of Interest

The authors have no conflicts of interest to declare.

Literature Cited

1. Herzenberg LA, Parks D, Sahaf B, Perez O, Roederer M, Herzenberg LA. The history and future of the fluorescence activated cell sorter and flow cytometry: A view from Stanford. Clin Chem 2002;48(10):1819–1827.
2. Pyne S, Hu X, Wang K, Rossin E, Lin T, Maier LM, et al. Automated high-dimensional flow cytometric data analysis. Proc Natl Acad Sci 2009;106(21): 8519–8524.
3. Bendall SC, Simonds EF, Qiu P, Amir ED, Krutzik PO, Finck R, Bruggner RV, Melamed R, Trejo A, Ornatsky OI, et al. Single-cell mass cytometry of differential a human hematopoietic continuum. Science 2011;687:687–697. https://doi.org/10.1126/science.1198704.
4. Stoeckius M, Hafemeister C, Stephenson W, Houck-Loomis B, Chattopadhyay PK, Swerdlow H, Satija R, Smibert P. Simultaneous epitope and transcriptome measurement in single cells. Nat Methods 2017;14(9):865–868. https://doi.org/10.1038/nmeth.4380.
5. Behbehani GK. Applications of mass cytometry in clinical medicine: The promise and perils of clinical CyTOF. Clin Lab Med 2017;37(4):945–964. https://doi.org/10.1016/j.cll.2017.07.010.
6. Kourou K, Exarchos TP, Exarchos KP, Karamouzis MV, Fotiadis DI. Machine learning applications in cancer prognosis and prediction. Comput Struct Biotechnol J 2014;13:8–17. https://doi.org/10.1016/j.csbj.2014.11.005.
7. Chester C, Maecker HT. Algorithmic tools for mining high-dimensional cytometry data. J Immunol 2015;195(3):773–779. https://doi.org/10.4049/jimmunol.1500633.
8. Kvistborg P, Gouttefangeas C, Aghaeepour N, Cazaly A, Chattopadhyay PK, Chan C, Eckl J, Finak G, Hadrup SR, Maecker HT, et al. Thinking outside the gate: Single-cell assessments in multiple dimensions. Immunity 2015;42(4):591–592. https://doi.org/10.1016/j.immuni.2015.04.006.
9. Aghaeepour N, Finak G, Hoos H, Mosmann TR, Brinkman R, Gottardo R, et al. Critical assessment of automated flow cytometry data analysis techniques. Nat Methods 2013;10(3):228–238. https://doi.org/10.1038/NMETH.2365.
10. Kimball AK, Oko LM, Bullock BL, Nemenoff RA, van Dyk LF, Clambey ET. A beginner's guide to analyzing and visualizing mass cytometry data. J Immunol 2018;200(1):3–22. https://doi.org/10.4049/jimmunol.1701494.
11. Olsen LR, Pedersen CB, Leipold MD, Maecker HT. Getting the most from your high-dimensional cytometry data. Immunity 2019;50(3):535–536. https://doi.org/10.1016/j.immuni.2019.02.015.
12. Troyanskaya O, Trajanoski Z, Carpenter A, Thrun S, Razavian N, Oliver N. Artificial intelligence and cancer. Nat Cancer 2020;1:149–152. https://doi.org/10.1038/s43018-020-0034-6.
13. Marx V. Machine learning, practically speaking. Nat Methods 2019;16:463–467. https://doi.org/10.1038/s41592-019-0432-9.
14. Keren L, Bosse M, Marquez D, Angoshtari R, Jain S, Varma S, Yang SR, Kurian A, van Valen D, West R, et al. A structured tumor-immune microenvironment in triple negative breast cancer revealed by multiplexed ion Beam imaging. Cell 2018; 174(6):1373–1387.e19. https://doi.org/10.1016/j.cell.2018.08.039.
15. Goltsev Y, Samusik N, Kennedy-Darling J, Bhate S, Hale M, Vazquez G, Black S, Nolan GP. Deep profiling of mouse splenic architecture with CODEX multiplexed imaging. Cell 2018;174(4):968–981.e15. https://doi.org/10.1016/j.cell.2018.07.010.
16. Polley MYC, Freidlin B, Korn EL, Conley BA, Abrams JS, McShane LM. Statistical and practical considerations for clinical evaluation of predictive biomarkers. J Natl Cancer Inst 2013;105(22):1677–1683. https://doi.org/10.1093/jnci/djt282.
17. Saeys Y, Inza I, Larrañaga P. A review of feature selection techniques in bioinformatics. Bioinformatics 2007;23(19):2507–2517. https://doi.org/10.1093/bioinformatics/btm344.
18. Chen YB, Cutler CS. Biomarkers for acute GVHD: Can we predict the unpredictable. Bone Marrow Transplant 2013;48(6):755–760. https://doi.org/10.1038/bmt.2012.143.
19. O'Neill K, Aghaeepour N, Špidlen J, Brinkman R. Flow cytometry bioinformatics. PLoS Comput Biol 2013;9(12):e1003365. https://doi.org/10.1371/journal.pcbi.1003365.
20. Jevremovic D, Olteanu H. Flow cytometry applications in the diagnosis of T/NK-cell lymphoproliferative disorders. Cytometry B Clin Cytom 2019;96B(2):99–115. https://doi.org/10.1002/cyto.b.21768.
21. Porwit A, Béné MC. Multiparameter flow cytometry applications in the diagnosis of mixed phenotype acute leukemia. Cytometry B Clin Cytom 2019;96B(3): 183–194. https://doi.org/10.1002/cyto.b.21783.
22. Mizrahi O, Ish Shalom E, Baniyash M, Klieger Y. Quantitative flow cytometry: Concerns and recommendations in clinic and research. Cytometry B Clin Cytom 2018;94B(2):211–218. https://doi.org/10.1002/cyto.b.21515.
23. Olsen LR, Leipold MD, Pedersen CB, Maecker HT. The anatomy of single cell mass cytometry data. Cytometry A 2018;95A(2):156–172. https://doi.org/10.1002/cyto.a.23621.
24. Beam AL, Kohane IS. Big data and machine learning in health care. JAMA 2018; 319(13):1317–1318. https://doi.org/10.1001/jama.2017.18391.
25. Ornatsky O, Bandura D, Baranov V, Nitz M, Winnik MA, Tanner S. Highly multiparametric analysis by mass cytometry. J Immunol Methods 2010;361(1–2):1–20.
26. Chen H, Lau MC, Wong MT, Newell EW, Poidinger M, Chen J. Cytofkit: A Bioconductor package for an integrated mass cytometry data analysis pipeline. PLoS Comput Biol 2016;12(9):e1005112. https://doi.org/10.1371/journal.pcbi.1005112.
27. Hartmann FJ, Bendall SC. Immune monitoring using mass cytometry and related high-dimensional imaging approaches. Nat Rev Rheumatol 2020;16:87–99. https://doi.org/10.1038/s41584-019-0338-z.
28. Jimenez-Carretero D, Ligos JM, Martínez-López M, Sancho D, Montoya MC. Flow cytometry data preparation guidelines for improved automated phenotypic analysis. J Immunol 2018;200(10):3319–3331. https://doi.org/10.4049/jimmunol.1800446.
29. Finck R, Simonds EF, Jager A, Krishnaswamy S, Sachs K, Fantl W, Pe'er D, Nolan GP, Bendall SC. Normalization of mass cytometry data with bead standards. Cytometry A 2013;83A(5):483–494. https://doi.org/10.1002/cyto.a.22271.
30. Hahne F, Khodabakhshi AH, Bashashati A, Wong CJ, Gascoyne RD, Weng AP, Seyfert-Margolis V, Bourcier K, Asare A, Lumley T, et al. Per-channel basis normalization methods for flow cytometry data. Cytometry A 2010;77A(2):121–131. https://doi.org/10.1002/cyto.a.20823.
31. Schuyler RP, Jackson C, Garcia-Perez JE, Baxter RM, Ogolla S, Rochford R, Ghosh D, Rudra P, Hsieh EWY. Minimizing batch effects in mass cytometry data. Front Immunol 2019;10:2367. https://doi.org/10.3389/fimmu.2019.02367.
32. Le Meur N, Rossini A, Gasparetto M, Smith C, Brinkman RR, Gentleman R. Data quality assessment of ungated flow cytometry data in high throughput experiments. Cytometry A 2007;71A(6):393–403. https://doi.org/10.1002/cyto.a.20396.
33. Zou H, Hastie T. Regularization and variable selection via the elastic net. J R Stat Soc Series B Stat Methodology 2005;67(2):301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x.
34. Shlens, J. (2014). A tutorial on principal component analysis. *arXiv e-prints*. Retrieved from http://arxiv.org/abs/1404.1100
35. Levine JH, Simonds EF, Bendall SC, Davis KL, Amir EAD, Tadmor MD, Litvin O, Fienberg HG, Jager A, Zunder ER, et al. Data-driven phenotypic dissection of AML reveals progenitor-like cells that correlate with prognosis. Cell 2015;162(1): 184–197. https://doi.org/10.1016/j.cell.2015.05.047.

36. Theunissen PMJ, Sedek L, De Haas V, Szczepanski T, Van Der Sluijs A, Mejstrikova E, et al. Detailed immunophenotyping of B-cell precursors in regenerating bone marrow of acute lymphoblastic leukaemia patients: Implications for minimal residual disease detection. Br J Haematol 2017;178(2):257–266. https://doi.org/10.1111/bjh.14682.
37. Theunissen, P., Mejstrikova, E., Sedek, L., Van Der Sluijs-Gelling, A. J., Gaipa, G., Bartels, M., … Van Der Velden, V. H. J. (2017). Standardized flow cytometry for highly sensitive MRD measurements in B-cell acute lymphoblastic leukemia. Blood, 129(3), 347–357. https://doi.org/10.1182/blood-2016-07-726307
38. Lever J, Krzywinski M, Altman N. Points of significance: Principal component analysis. Nat Methods 2017;14(7):641–642. https://doi.org/10.1038/nmeth.4346.
39. Amir EAD, Davis KL, Tadmor MD, Simonds EF, Levine JH, Bendall SC, et al. ViSNE enables visualization of high dimensional single-cell data and reveals phenotypic heterogeneity of leukemia. Nat Biotechnol 2013;31(6):545–552. https://doi.org/10.1038/nbt.2594.
40. van der Maaten L, Hinton GE. Visualizing Data using t-SNE. J Mach Learn Res 2008;164(2210):10.
41. Wattenberg M Fernanda Viégas, Ian Johnson, How to Use t-SNE Effectively, Distill, 2016. http://doi.org/10.23915/distill.00002
42. van Unen V, Höllt T, Pezzotti N, Li N, Reinders MJT, Eisemann E, Koning F, Vilanova A, Lelieveldt BPF. Visual analysis of mass cytometry data by hierarchical stochastic neighbour embedding reveals rare cell types. Nat Commun 2017;8:1740. https://doi.org/10.1038/s41467-017-01689-9.
43. Belkina AC, Ciccolella CO, Anno R, Halpert R, Spidlen J, Snyder-Cappione JE. Automated optimized parameters for T-distributed stochastic neighbor embedding improve visualization and analysis of large datasets. Nat Commun 2019;10:5415. https://doi.org/10.1038/s41467-019-13055-y.
44. Chew V, Lai L, Pan L, Lim CJ, Li J, Ong R, Chua C, Leong JY, Lim KH, Toh HC, et al. Delineation of an immunosuppressive gradient in hepatocellular carcinoma using high-dimensional proteomic and transcriptomic analyses. Proc Natl Acad Sci 2017;114(29):E5900–E5909. https://doi.org/10.1073/pnas.1706559114.
45. Lowther DE, Goods BA, Lucca LE, Lerner BA, Raddassi K, van Dijk D, Hernandez AL, Duan X, Gunel M, Coric V, et al. PD-1 marks dysfunctional regulatory T cells in malignant gliomas. JCI Insight 2016;1(5):1–15. https://doi.org/10.1172/jci.insight.85935.
46. Ferrell PB, Diggins KE, Polikowsky HG, Mohan SR, Seegmiller AC, Irish JM. High-dimensional analysis of acute myeloid leukemia reveals phenotypic changes in persistent cells during induction therapy. PLoS One 2016;11(4):e0153207. https://doi.org/10.1371/journal.pone.0153207.
47. Bandyopadhyay S, Fowles JS, Yu L, Fisher DAC, Oh ST. Identification of functionally primitive and immunophenotypically distinct subpopulations in secondary acute myeloid leukemia by mass cytometry. Cytometry B Clin Cytom 2019;96B(1):46–56. https://doi.org/10.1002/cyto.b.21743.
48. McInnes, L., Healy, J., Melville, J. (2018). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. Retrieved from http://arxiv.org/abs/1802.03426
49. Becht E, McInnes L, Healy J, Dutertre CA, Kwok IWH, Ng LG, Ginhoux F, Newell EW. Dimensionality reduction for visualizing single-cell data using UMAP. Nat Biotechnol 2019;37(1):38–47. https://doi.org/10.1038/nbt.4314.
50. Diaz-Papkovich, A., Anderson-Trocme, L., & Gravel, S. (2019). Revealing multi-scale population structure in large cohorts. BioRxiv. https://doi.org/10.1101/423632
51. Becht, E., Dutertre, C., Kwok, I., Ng, L., Ginhoux, F. (2018). Evaluation of UMAP as an alternative to t-SNE for single-cell data. BioRxiv.
52. Oetjen KA, Lindblad KE, Goswami M, Gui G, Dagur PK, Lai C, Dillon LW, McCoy JP, Hourigan CS. Human bone marrow assessment by single-cell RNA sequencing, mass cytometry, and flow cytometry. JCI Insight 2018;3(23):e124928. https://doi.org/10.1172/jci.insight.124928.
53. Coenen A, Pearce A. Understanding UMAP. Google PAIR blog, 2020. https://pair-code.github.io/understanding-umap/
54. Shibata M, Hoque MO. Targeting cancer stem cells: A strategy for effective eradication of cancer. Cancer 2019;11(5):732. https://doi.org/10.3390/cancers11050732.
55. Gay L, Baker A-M, Graham TA. Tumour Cell Heterogeneity. F1000Res 2016;5:238. https://doi.org/10.12688/f1000research.7210.1.
56. Murtagh F, Contreras P. Algorithms for hierarchical clustering: An overview. Wiley Interdiscip Rev Data Min Knowl Discov 2012;2(1):86–97. https://doi.org/10.1002/widm.53.
57. Yang Y. Temporal data clustering. Temporal Data Mining Via Unsupervised Ensemble Learning, 2016; p. 19–34). Amsterdam, Netherlands: Elsevier. https://doi.org/10.1016/b978-0-12-811654-8.00003-8.
58. Johnson SC. Hierarchical clustering schemes. Psychometrika 1967;32(3).241–254.
59. Qiu P, Simonds EF, Bendall SC, Gibbs KD, Bruggner RV, Linderman MD, et al. Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE. Nat Biotechnol 2011;29(10):886–891. https://doi.org/10.1038/nbt.1991.
60. Qiu P. Toward deterministic and semiautomated SPADE analysis. Cytometry A 2017;91A(3):281–289. https://doi.org/10.1002/cyto.a.23068.
61. Siska PJ, Beckermann KE, Mason FM, Andrejeva G, Greenplate AR, Sendor AB, Chiang YCJ, Corona AL, Gemta LF, Vincent BG, et al. Mitochondrial dysregulation and glycolytic insufficiency functionally impair CD8 T cells infiltrating human renal cell carcinoma. JCI Insight 2017;2(12):e93411. https://doi.org/10.1172/jci.insight.93411.
62. Van Gassen S, Callebaut B, Van Helden MJ, Lambrecht BN, Demeester P, Dhaene T, Saeys Y. FlowSOM: Using self-organizing maps for visualization and interpretation of cytometry data. Cytometry A 2015;87A(7):636–645. https://doi.org/10.1002/cyto.a.22625.
63. Doxie DB, Greenplate AR, Gandelman JS, Diggins KE, Roe CE, Dahlman KB, Sosman JA, Kelley MC, Irish JM. BRAF and MEK inhibitor therapy eliminates nestin-expressing melanoma cells in human tumors. Pigment Cell Melanoma Res 2018;31(6):708–719. https://doi.org/10.1111/pcmr.12712.
64. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning, New York, NY: Springer, 2009.
65. Weber LM, Robinson MD. Comparison of clustering methods for high-dimensional single-cell flow and mass cytometry data. Cytometry A 2016;89A(12):1084–1096. https://doi.org/10.1002/cyto.a.23030.
66. Aghaeepour N, Nikolic R, Hoos HH, Brinkman RR. Rapid cell population identification in flow cytometry data. Cytometry A 2011;79A(1):6–13. https://doi.org/10.1002/cyto.a.21007.
67. Blondel VD, Guillaume J-L, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. J Stat Mech 2008;2008(10):P10008. https://doi.org/10.1088/1742-5468/2008/10/P10008.
68. Newman AM, Liu CL, Green MR, Gentles AJ, Feng W, Xu Y, et al. Robust enumeration of cell subsets from tissue expression profiles. Nat Methods 2015;12:453–457. https://doi.org/10.1038/nmeth.3337.
69. Emmons S, Kobourov S, Gallant M, Börner K. Analysis of network clustering algorithms and cluster quality metrics at scale. PLoS One 2016;11(7):1–18. https://doi.org/10.1371/journal.pone.0159161.
70. Real R, Vargas JM. The probabilistic basis of Jaccard's index of similarity. Syst Biol 1996;45(3):380–385. https://doi.org/10.1093/sysbio/45.3.380.
71. Chevrier S, Levine JH, Zanotelli VRT, Silina K, Schulz D, Bacac M, Ries CH, Ailles L, Jewett MAS, Moch H, et al. An immune atlas of clear cell renal cell carcinoma. Cell 2017;169(4):736–749.e18. https://doi.org/10.1016/j.cell.2017.04.016.
72. Lavin Y, Kobayashi S, Leader A, Amir E a D, Elefant N, Bigenwald C, et al. Innate immune landscape in early lung adenocarcinoma by paired single-cell analyses. Cell 2017;169(4):750–765.e17. https://doi.org/10.1016/j.cell.2017.04.014.
73. Zhu YP, Padgett L, Dinh HQ, Marcovecchio P, Blatchley A, Wu R, Ehinger E, Kim C, Mikulski Z, Seumois G, et al. Identification of an early Unipotent neutrophil progenitor with pro-tumoral activity in mouse and human bone marrow. Cell Rep 2018;24(9):2329–2341.e8. https://doi.org/10.1016/j.celrep.2018.07.097.
74. Andrews TS, Hemberg M. Identifying cell populations with scRNASeq. Mol Asp Med 2018;59:114–122. https://doi.org/10.1016/j.mam.2017.07.002.
75. Samusik N, Good Z, Spitzer MH, Davis KL, Nolan GP. Automated mapping of phenotype space with single-cell data. Nat Methods 2016;13(6):493–496. https://doi.org/10.1038/nmeth.3863.
76. Neuberg D, Cader FZ, Chapuy B, Armand P, Rodig SJ, Schackmann RCJ, et al. Mass cytometry of Hodgkin lymphoma reveals a CD4 + exhausted T-effector and T-regulatory cell rich microenvironment. Blood 2018;132(8):825–836. https://doi.org/10.1182/blood-2018-04-843714.
77. Gonzalez VD, Samusik N, Chen TJ, Savig ES, Aghaeepour N, Quigley DA, Huang YW, Giangarrà V, Borowsky AD, Hubbard NE, et al. Commonly occurring cell subsets in high-grade serous ovarian tumors identified by single-cell mass cytometry. Cell Rep 2018;22(7):1875–1888. https://doi.org/10.1016/j.celrep.2018.01.053.
78. Chen G, Ning B, Shi T. Single-cell RNA-Seq technologies and related computational data analysis. Front Genet 2019;10:1–13. https://doi.org/10.3389/fgene.2019.00317.
79. Bruggner RV, Bodenmiller B, Dill DL, Tibshirani RJ, Nolan GP. Automated identification of stratifying signatures in cellular subpopulations. Proc Natl Acad Sci 2014;111(26):E2770–E2777. https://doi.org/10.1073/pnas.1408792111.
80. Tibshirani R. Regression shrinkage and selection via the lasso. J R Stat Soc B Methodol 1996;58(1):267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x.
81. Wang S, Zhu J. Improved centroids estimation for the nearest shrunken centroid classifier. Bioinformatics 2007;23(8):972–979. https://doi.org/10.1093/bioinformatics/btm046.
82. Spitzer MH, Carmi Y, Reticker-Flynn NE, Kwek SS, Madhireddy D, Martins MM, Gherardini PF, Prestwood TR, Chabon J, Bendall SC, et al. Systemic immunity is required for effective cancer immunotherapy. Cell 2017;168(3):487–502.e15. https://doi.org/10.1016/j.cell.2016.12.022.
83. Spitzer MH, Gherardini PF, Fragiadakis GK, Bhattacharya N, Yuan RT, Hotson AN, Finck R, Carmi Y, Zunder ER, Fantl WJ, et al. An interactive reference framework for modeling a dynamic immune system. Science 2015;349(6244):1259425. https://doi.org/10.1126/science.1259425.
84. Tusher VG, Tibshirani R, Chu G. Significance analysis of microarrays applied to the ionizing radiation response. PNAS 2001;98(9):5116–5121.
85. Avanzi MP, Yeku O, Li X, Wijewarnasuriya DP, van Leeuwen DG, Cheung K, Park H, Purdon TJ, Daniyan AF, Spitzer MH, et al. Engineered tumor-targeted T cells mediate enhanced anti-tumor efficacy both directly and through activation of the endogenous immune system. Cell Rep 2018;23(7):2130–2141. https://doi.org/10.1016/j.celrep.2018.04.051.
86. Chen X, Sun X, Hoshida Y. Survival analysis tools in genomics research. Hum Genomics 2014;8(1):1–5. https://doi.org/10.1186/s40246-014-0021-z.
87. Cox DR. Regression models and life-tables. Biometrika 1972;45(3):562–565.
88. Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. J Stat Softw 2010;33(1):1–22. http://www.jstatsoft.org/v33/i01/.
89. Robinson MD, McCarthy DJ, Smyth GK. edgeR: A Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics 2010;26(1):139–140. https://doi.org/10.1093/bioinformatics/btp616.
90. Kuhn M. Building predictive models in R using the caret package. J Stat Softw 2008;28(5):1–26.

91. Good Z, Sarno J, Jager A, Samusik N, Aghaeepour N, Simonds EF, et al. Single-cell developmental classification of B cell precursor acute lymphoblastic leukemia at diagnosis reveals predictors of relapse. Nat Med 2018;24:474–483. https://doi.org/10.1038/nm.4505.
92. Wickham H, Averick M, Bryan J, Chang W, D'Agostino McGowan L, François R, Grolemund G, Hayes A, Henry L, Hester J, et al. Welcome to the Tidyverse. J Open Source Softw 2019;4(43):1686. https://doi.org/10.21105/joss.01686.
93. Kuhn M, Johnson K. Applied Predictive Modeling. New York: Springer, 2013.
94. Diggins KE, Greenplate AR, Leelatian N, Wogsland CE, Irish JM. Characterizing cell subsets using marker enrichment modeling. Nat Methods 2017;14(3):275–278. https://doi.org/10.1038/nmeth.4149.
95. Aghaeepour, N., Simonds, E. F., Knapp, D. J. H. F., Bruggner, R. V., Sachs, K., Culos, A., … Nolan, G. P. (2018). GateFinder: Projection-based gating strategy optimization for flow and mass cytometry. Bioinformatics, 34(23), 4131–4133. https://doi.org/10.1093/bioinformatics/bty430
96. Becht E, Simoni Y, Coustan-Smith E, Evrard M, Cheng Y, Ng LG, Campana D, Newell EW. Reverse-engineering flow-cytometry gating strategies for phenotypic labelling and high-performance cell sorting. Bioinformatics 2019;35(2):301–308. https://doi.org/10.1093/bioinformatics/bty491.
97. Hu Z, Glicksberg BS, Butte AJ. Robust prediction of clinical outcomes using cytometry data. Bioinformatics 2019;35(7):1197–1203. https://doi.org/10.1093/bioinformatics/bty768.
98. Seiler, C., Kronstad, L. M., Simpson, L. J., Gars, M. Le, Vendrame, E., Blish, C. A., & Holmes, S. (2019). Uncertainty quantification in multivariate mixed models for mass cytometry data, (Cd). Retrieved from http://arxiv.org/abs/1903.07976
99. Finak G, Frelinger J, Jiang W, Newell EW, Ramey J, Davis MM, Kalams SA, De Rosa SC, Gottardo R. OpenCyto: An open source infrastructure for scalable, robust, reproducible, and automated, end-to-end flow cytometry data analysis. PLoS Comput Biol 2014;10(8):e1003806. https://doi.org/10.1371/journal.pcbi.1003806.
100. Lo K, Hahne F, Brinkman RR, Gottardo R. flowClust: A Bioconductor package for automated gating of flow cytometry data. BMC Bioinformatics 2009;10:145. https://doi.org/10.1186/1471-2105-10-145.
101. Malek M, Taghiyar MJ, Chong L, Finak G, Gottardo R, Brinkman RR. flowDensity: reproducing manual gating of flow cytometry data by automated density-based cell population identification. Bioinformatics 2015;31(4):606–607. https://doi.org/10.1093/bioinformatics/btu677.
102. Lux M, Brinkman RR, Chauve C, Laing A, Lorenc A, Abeler-Dörner L, Hammer B. flowLearn: Fast and precise identification and quality checking of cell populations in flow cytometry. Bioinformatics 2018;34(13):2245–2253. https://doi.org/10.1093/bioinformatics/bty082.
103. Li H, Shaham U, Stanton KP, Yao Y, Montgomery RR, Kluger Y. Gating mass cytometry data by deep learning. Bioinformatics 2017 Nov 1;33(21):3423–3430. https://doi.org/10.1093/bioinformatics/btx448.
104. Buldini B, Maurer-Granofszky M, Varotto E, Dworzak MN. Flow-cytometric monitoring of minimal residual disease in pediatric patients with acute myeloid leukemia: Recent advances and future strategies. Front Pediatr 2019 Oct 11;7:412. https://doi.org/10.3389/fped.2019.00412.
105. Gupta A, Harrison PJ, Wieslander H, Pielawski N, Kartasalo K, Partel G, et al. Deep learning in image cytometry: A review. Cytometry A 2019;95A(4):366–380. https://doi.org/10.1002/cyto.a.23701.
106. Amodio M, van Dijk D, Srinivasan K, Chen WS, Mohsen H, Moon KR, Campbell A, Zhao Y, Wang X, Venkataswamy M, et al. Exploring single-cell data with deep multitasking neural networks. Nat Methods 2019;16:1139–1145. https://doi.org/10.1038/s41592-019-0576-7.

Glossary

Introduction and Overview

Machine learning (ML) -
In general, a poorly-defined term. Here, it is used to refer to the field of statistics that involves training any of a large family of models in order to best find, classify, or predict patterns in data based on a carefully selected set of assumptions.

Supervised learning -
ML approaches that explicitly associate observations' input variables (often called "predictors") with outcome variables such as survival or treatment response.

Unsupervised learning -
ML approaches that organize observations - either into groups or along a continuum - based solely on their input features and without access to outcome labels of any kind.

Dimensionality Reduction

A family of analytical approaches in which high-dimensional data are embedded in lower-dimensional space. Commonly used to visualize data with greater than 3 dimensions.

Factor loadings -
The correlations between each of a dataset's original variables and each of its principal components (PCs); equivalently, the projection of each original variable onto each PC. Can be useful for interpreting the information represented by each PC.

Nonlinearity -
In most cases, a dataset is described as "nonlinear" when its variables are not well-described as a linear function of its other variables. A dataset may also be described as "nonlinear" when an *external* outcome variable related to the dataset is not easily computed as a linear function of its variables. Often, this complexity is a result of interaction terms between variables and/or higher-order polynomial/power-law relationships.

Singular Value Decomposition (SVD) -
A matrix factorization method that, when computed, can be used to find the best low-dimensional representation for a dataset under the assumption that the data's variables have linear relationships with one another. Can be calculated rapidly for even very large matrices (i.e., those that represent very large datasets).

Clustering

A type of unsupervised learning in which observations are placed into groups such that similar observations are grouped together and dissimilar observations are not. Often applied to single-cell data to detect distinct cellular subpopulations.

Jaccard Coefficient -
A measure of how "connected" two nodes in a graph are to one another. Specifically, the Jaccard Coefficient is the number of neighbors that two nodes share divided by the total number of neighbors of both nodes. For two nodes with sets of neighbors A and B, the Jaccard Coefficient is given by $\frac{|A \cap B|}{|A \cup B|}$.

Minimum-Spanning Tree (MST) -
A *spanning tree* is a type of graph in which all nodes are connected to one another without any loops or breaks in continuity. A *minimum spanning tree*, then, is the spanning tree with the smallest total edge length of all possible spanning trees for a given set of nodes.

Self-organizing map (SOM) -
A type of artificial neural network capable of dimensionality reduction and clustering of high-dimensional data. Computed using a series of recursive, non-parametric regression computational steps.

Prediction and Correlative Biology

Generalized Linear Models (GLMs) -
A class of statistical models in which linear combinations of input variables are transformed via a "link function" that allows them to predict nonlinear response variables. In other words, GLMs are an extension of linear regression such that nonlinear relationships can be predicted. For example, the log-odds (or "logit") link function allows the GLM framework to "extend" linear regression, which can only explicitly predict continuous outcomes, to logistic regression, which can be used to predict the probability of an observation belonging to one of two classes.

Feature selection -
A term referring to a general analytical approach in which a subset of a dataset's input variables are identified (manually or automatically) as the most important for predicting or explaining an outcome of interest.

Significance Analysis of Microarrays (SAM) -
An analytical approach in which differences in marker expression distributions are detected by permuting sample labels randomly in order to define a null distribution of "expected" differences resulting from chance variation. Originally published in 2001 to analyze microarrays, this approach has since been applied to differential expression analyses on a variety of transcript- and protein-level data.

Miscellaneous

Deep learning -
A subfield within ML in which artificial neural networks are used to solve both supervised and unsupervised learning problems. An area of rapid growth in bioinformatics.

Distance metrics -
We discuss several distance metrics in the text. For the n-dimensional vectors x and y, these distances are given by…

Manhattan (L1-norm): $\sum_i^n |x_i - y_i|$. Used in Lasso regularization and elastic net.

Euclidean (L2-norm): $\sum_i^n \sqrt{x_i^2 - y_i^2}$. Most commonly-used distrance metric across single-cell data types. Used by many algorithms, including PhenoGraph.

Pearson: $1 - r_{xy}$, where r_{xy} is the Pearson correlation coefficient between x and y

Cosine: $1 - \frac{\sum_i^n x_i y_i}{\sqrt{\sum_i^n x_i^2}\sqrt{\sum_i^n y_i^2}}$. Represents the angle between the vectors x and y, and is invariant to their size. Used by Scaffold.

Mahalanobis: $\sqrt{(x-y)S^{-1}(x-y)}$, where S is the covariance matrix for the distribution that contains x and y. Often thought of as a "multidimensional Z-score."

Hyperparameters (tuning parameters) -
Numeric values that influence how a model is computed, but are not estimated directly from input data. Generally speaking, users specify hyperparameter values through trial and error or through an exhaustive "grid search" of many values.

Cytometry Part A
Author Checklist: MIFlowCyt-Compliant Items

Requirement	Please Include Requested Information
1.1. Purpose	The original purpose of this dataset was to develop a predictive model of post-treatment relapse in pediatric B-cell progenitor acute lymphoblastic leukemia (BCP-ALL) based on cells' developmental phenotype. See https://doi.org/10.1038/nm.4505
1.2. Keywords	BCP-ALL, mass cytometry, leukemia
1.3. Experiment variables	Conditional variables: Diagnosis, Relapse Manipulated Variables: Stimulation
1.4. Organization name and address	Stanford University School of Medicine, 291 Campus Drive, Stanford, CA 94305
1.5. Primary contact name and email address	Kara Davis, DO Email: kardavis@stanford.edu
1.6. Date or time period of experiment	06/4/2014 to 07/21/2015
1.7. Conclusions	We found that aberrant and unresponsive B-cell receptor signaling at the pro-BII/pre-BI stage in diagnostic BCP-ALL samples portend relapse of the disease.
1.8. Quality control measures	Four different bone marrow healthy donors were used across each CyTOF run and all together as control
2.1.1.1. (2.1.2.1., 2.1.3.1.) Sample description	Bone marrow cryopreserved cells at the time of diagnosis (n=60) were analyzed. In addition to that, bone marrow cryopreserved cells from relapse timepoint were also processed for 7 patients.
2.1.1.2. Biological sample source description	De-identified samples from pediatric patients with BCP-ALL were obtained under informed consent from the Lucile Packard Children's Hospital at Stanford (Stanford, CA, USA; Ph+ samples, n = 9) and the Pediatric Clinic University of Milan Bicocca (Monza, Italy; n = 51) for a total of 60 primary diagnostic patient samples. Use of these samples was approved by the Institutional Review Boards at both institutions.
2.1.1.3. Biological sample source organism description	Samples were obtained from mononuclear cells isolated from bone marrow aspirates
2.1.2.2. Environmental sample location	Cryopreserved mononuclear cells were stored in liquid nitrogen tanks at Pediatric Clinic University of Milan Bicocca (Monza, Italy) and at Lucile Packard Children's Hospital at Stanford (Stanford, CA, USA).
2.3. Sample treatment description	Viable preserved bone marrow cells were thawed, rested for 30 minutes at 37C and stained for viability with cisplatin. Following viability staining, cells were perturbed under the following conditions: treatment with pervanadate, IL-7, thymic stromal lymphopoietin (TSLP), dasatinib, BEZ-235 or tofacitinib,or by BCR crosslinking.
2.4. Fluorescence reagent(s) description	N/A
3.1. Instrument manufacturer	DVS Sciences (Fluidigm)
3.2. Instrument model	CyTOF-1
3.3. Instrument configuration and settings	CyTOF-1 software Ver 5.1.648 (data 6/42014-5/19/2015) CyTOF-1 software Ver 6.0.626 partial upgrade for data

	amplifier, computer, and software (data 5/20/2015-7/21/2015)
4.1. List-mode data files	*We recommend all authors to submit their data files to http://flowrepository.org and to make them available for the peer-review process. If you have done so, please let us know by inserting the following codes (replace the red text): 1) The link for peer-review process: http://flowrepository.org/id/RvFrxxxxxx (copy and paste the code). This link will only be shared with reviewers of your manuscript. 2) The repository identifier: http://flowrepository.org/id/FR-FCM-xxxx (copy and paste the code). This link will be made publicly accessible after the paper is published.
4.2. Compensation description	N/A
4.3. Data transformation details	Single-cell protein expression data were extracted using Bioconductor software (http://www.bioconductor.org) and transformed using the inverse hyperbolic sine (arsinh) function with a cofactor of 5. To control for batch effects among barcoding plates, we performed percentile normalization using the healthy reference BM sample(s) that were included within each plate
4.4.1. Gate description	Single cells were gated using Cytobank software based on event length and 191Ir/193Ir DNA content. Following single-cell gating, live non-apoptotic cells were gated based on cleaved poly(ADP-ribose) polymerase (cPARP) and 195Pt content. Platelets and erythrocytes were excluded by gating on CD61 and CD235a, respectively. The remaining fraction was gated to exclude T cells and myeloid cells on the basis of CD3e and of CD33 and CD16 expression, respectively. After further exclusion of CD38high plasma cells, the remaining fraction was defined as lineage-negative blasts. Further analysis was applied to this Lin− B+ fraction unless otherwise noted.
4.4.2. Gate statistics	Percentage of cells positive for key predictive cellular features in pro-BII and pre-BI cells were calculated for downstream analysis.
4.4.3. Gate boundaries	Calculation of the percentage of positive cells for each phosphorylated protein was based on a mass cytometry cutoff of ≥10 counts.

Notes

Feel free to use more space than allocated.
You can embed graphics/figures in this document, if needed.
Please make sure to save the document in Microsoft Word version 2003 or older, before uploading to ScholarOne Manuscripts. When uploading this checklist to ScholarOne Manuscripts, please choose the "Supplementary Material for Review" category.
Please note that if your paper is accepted, the checklist will be published as an Online Supporting Information.

For any questions, please contact the Cytometry Part A editorial office at Cytometrya@wiley.com.