bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Joint probabilistic modeling of paired transcriptome and proteome measurements in single cells

Adam Gayoso [1, *], Zoë Steier [2, *], Romain Lopez [3], Jeffrey Regier [4], Kristopher L Nazor [5], Aaron Streets [1, 2, 6, †], and Nir Yosef [1, 3, 6, 7, †]

[1] Center for Computational Biology,
University of California, Berkeley,
Berkeley, CA, USA

[2] Department of Bioengineering,
University of California, Berkeley,
Berkeley, CA, USA

[3] Department of Electrical Engineering and Computer Sciences,
University of California, Berkeley,
Berkeley, CA, USA

[4] Department of Statistics,
University of Michigan, Ann Arbor,
Ann Arbor, MI, USA

[5] BioLegend, Inc.,
San Diego, CA, USA

[6] Chan Zuckerberg Biohub,
San Francisco, CA, USA

[7] Ragon Institute of MGH, MIT and Harvard

* These authors contributed equally.

† Corresponding author: `{astreets, niryosef}@berkeley.edu`

Abstract

The paired measurement of RNA and surface protein abundance in single cells with CITE-seq is a promising approach to connect transcriptional variation with cell phenotypes and functions. However, each data modality exhibits unique technical biases, making it challenging to conduct a joint analysis and combine these two views into a unified representation of cell state. Here we present Total Variational Inference (totalVI), a framework for the joint probabilistic analysis of paired RNA and protein data from single cells. totalVI probabilistically represents the data as a composite of biological and technical factors such as limited sensitivity of the RNA data, background in the protein data, and batch effects. To evaluate totalVI, we performed CITE-seq on immune cells from murine spleen and lymph nodes with biological replicates and with different antibody panels measuring over 100 surface proteins. With this dataset, we demonstrate that totalVI provides a cohesive solution for common analysis tasks like the integration of datasets with matched or unmatched protein panels, dimensionality reduction, clustering, evaluation of correlations between molecules, and differential expression testing. totalVI enables scalable, end-to-end analysis of paired RNA and protein data from single cells and is available as open-source software.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

1 Introduction

The advance of technologies for quantitative, high-throughput measurement of the molecular composition of single cells is continuously expanding our understanding of cell ontology, state, and function [1–3]. Flow cytometry has set the gold standard for classifying cell types and phenotypic states by quantifying the abundance of small panels of marker proteins on the surface of each cell. More recently, unbiased whole-transcriptome profiling with single-cell RNA sequencing (scRNA-seq) has enabled more comprehensive characterization of cell types [4] and a mechanistic understanding of molecular processes in single cells [5]. Flow cytometry and scRNA-seq offer different views of biological systems at the single-cell level, each with unique insights and limitations. In flow cytometry, for example, fluorescent detection of protein abundance limits how many proteins can be probed simultaneously, whereas scRNA-seq only provides a proxy for the functional information contained in the proteome that does not necessarily reflect protein abundance [6, 7]. The development of techniques such as CITE-seq [8, 9] for paired measurement of RNA and surface proteins from the same cell now provides the opportunity to jointly leverage both types of information. An important feature of CITE-seq is that it quantifies protein abundance by sequencing barcoded antibodies. This significantly increases the number of proteins that can be measured simultaneously relative to flow cytometry and provides the potential to approach proteome-wide profiling of single cells.

Such multi-omic single-cell measurements have the power to provide many new biological insights. First, connecting the extensive literature on cell surface markers with whole-transcriptome profiles could facilitate more consistent and robust annotations of cell types. More fundamentally, having access to a comprehensive molecular profile of each cell could enable a more nuanced, multifaceted understanding of its state [10, 11]. Furthermore, combining these two modalities presents new opportunities such as identifying novel surface markers for new or known cell types, estimating the rates of protein processing [12], and exploring how signals may be received through the cell's outer membrane and transmitted to produce transcriptional changes. A joint analysis that combines the transcriptomic and proteomic views of a cell could help achieve these goals by using information from both molecular measurements to inform all downstream analysis.

However, combining the information contained in both molecular measurements poses multiple challenges. For example, while the RNA and protein counts observed in CITE-seq data reflect biological factors related to the cell's underlying state, both measurements contain technical biases that are specific to each modality, making normalization difficult. The RNA data suffer from noise, limited sensitivity, and variation in sequencing depth – factors that have been addressed by a flourishing body of probabilistic methods [13–16]. The protein data, while less sparse than RNA, tend to be obscured by noise and background, which is caused by the detection of ambient or non-specifically bound antibodies. Next, there is the modeling challenge of combining the distinct sets of information and constructing a unified representation for each cell [17, 18]. Finally, there is the challenge of linking a joint data representation to downstream analysis tasks such as dimensionality reduction, clustering, differential expression, and dataset integration. As large-scale community efforts such as the Human Cell Atlas (HCA; [4]) begin to include CITE-seq datasets, these tasks become increasingly challenging because different experiments may use different antibody panels. This is problematic because algorithms designed for these tasks often require direct correspondence of features (genes and proteins) between datasets, which would entail computationally intersecting antibody panels and subsequently, a loss of some or all of the protein information.

These challenges present a need for a scalable, joint statistical model of RNA and protein measurements. Most CITE-seq studies thus far have limited their analysis such that variability between cells was represented by only one modality, while the other modality served to validate and aid with interpretation post-hoc [19–21]. For instance, two recent studies [19, 20] used the RNA data to cluster cells, and then the protein data to derive cluster annotations. Furthermore, statistical methods that have been applied to the protein data do not adequately address the technical factors inherent to this measurement.

Here, we present totalVI (Total Variational Inference), a deep generative model for paired RNA-protein data analysis that addresses these challenges simultaneously. totalVI learns a joint probabilistic representation of RNA and protein measurements that aims to account for the distinct noise and technical biases of each modality as well as batch effects. For RNA, totalVI uses a modeling strategy similar to our previous work on scRNA-seq (scVI; [13]). For proteins, totalVI introduces a new model that separates the protein signal into background and foreground components. It then estimates

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

the probability that the counts of a protein observed in any given cell reflect protein abundance on the cell surface rather than technical artifacts caused by ambient antibodies or non-specific binding. The probabilistic representations learned by totalVI are built on a joint low-dimensional view of the RNA and protein data that is derived using neural networks. This joint view captures the biological variability between cells and can be interpreted as representing the underlying state of each cell. As such, this single low-dimensional representation of cell state reflects the coordinated regulation of transcription and translation that produces both molecules in the same cell.

The totalVI joint probabilistic model provides a cohesive solution for common analysis tasks such as dimensionality reduction, clustering, evaluation of correlations between genes and/or proteins, and differential expression testing. Prior to conducting these procedures, totalVI can be used for dataset integration, even in the case of partially- or non-overlapping protein panels, which result in "missing" protein entries. In addition to the integration of two or more input datasets, totalVI can impute the expression of proteins that are missing in certain samples, including ones with no measured proteins such as those from scRNA-seq experiments. Finally, while totalVI is based on deep neural networks, it provides a way to relate the coordinates of its latent dimensions to the expression of specific genes and proteins, thus affording interpretability and highlighting interesting trends in the data. In each of these tasks, totalVI accounts for batch effects, protein background, and other nuisance factors to enable more accurate analysis.

To evaluate the performance of totalVI, we conducted a series of CITE-seq experiments on murine spleen and lymph nodes that were designed to include complications like biological replicates, different protein panels, and up to 208 measured proteins per experiment. Using this data along with several public datasets, we demonstrated that totalVI effectively decouples protein foreground and background, integrates datasets with either matched or unmatched protein panels, imputes the measurements of proteins that were not included in one of the panels, and detects differential expression of both RNA and proteins. We demonstrated that totalVI compares favorably to the state of the art methods in each of these tasks (in cases where such a method is available), and that it scales to large dataset sizes. We applied totalVI to characterize the heterogeneity of B cells in the spleen and lymph nodes with paired RNA and protein information. totalVI and its downstream analysis procedures are available as open source software, which can be readily used as an end-to-end analytical solution for the increasingly common CITE-seq technology.

2 Results

2.1 The totalVI model

totalVI uses a probabilistic latent variable model [22] to represent the uncertainty in the observed RNA and protein counts from a CITE-seq experiment as a composite of biological and technical sources of variation. The input to totalVI consists of the matrices of RNA and protein unique molecular identifier (UMI) counts (Figure 1a). Categorical covariates such as experimental batch or patient are optional inputs used for integrating datasets, and referred to henceforth as "batch". totalVI can integrate datasets with different antibody panels (including integration with standard scRNA-seq), and use its probabilistic model to impute any missing protein measurements.

The output of totalVI consists of two components that can be used for downstream analysis. The first component encodes each cell as a distribution in an unobserved (latent) low-dimensional space that represents the information contained in the RNA and protein data and controls for their respective noise properties. The second component provides a way to estimate the parameters of the distributions that underlie the observed RNA and protein measurements (i.e., likelihoods) given a cell's latent representation. These distributions explicitly account for nuisance factors in the observed data such as sequencing depth, protein background, and batch effects (Figure 1b).

To model the RNA data, totalVI posits that for each gene g and cell n the number of observed transcripts, x_{ng}, follows a negative binomial distribution, which has been shown to adequately handle the limited sensitivity and over-dispersion that are characteristic of this data [23]. The mean of the negative binomial, $\ell_n \rho_{ng}$, depends on the cell's batch, denoted s_n, and two latent variables. The first variable, ℓ_n, represents cell size and sequencing depth and follows a lognormal distribution. The second variable, z_n, is a low-dimensional vector (here, 20 dimensions), which follows a logistic normal distribution and can be interpreted as the underlying biological state of the cell. A neural network

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

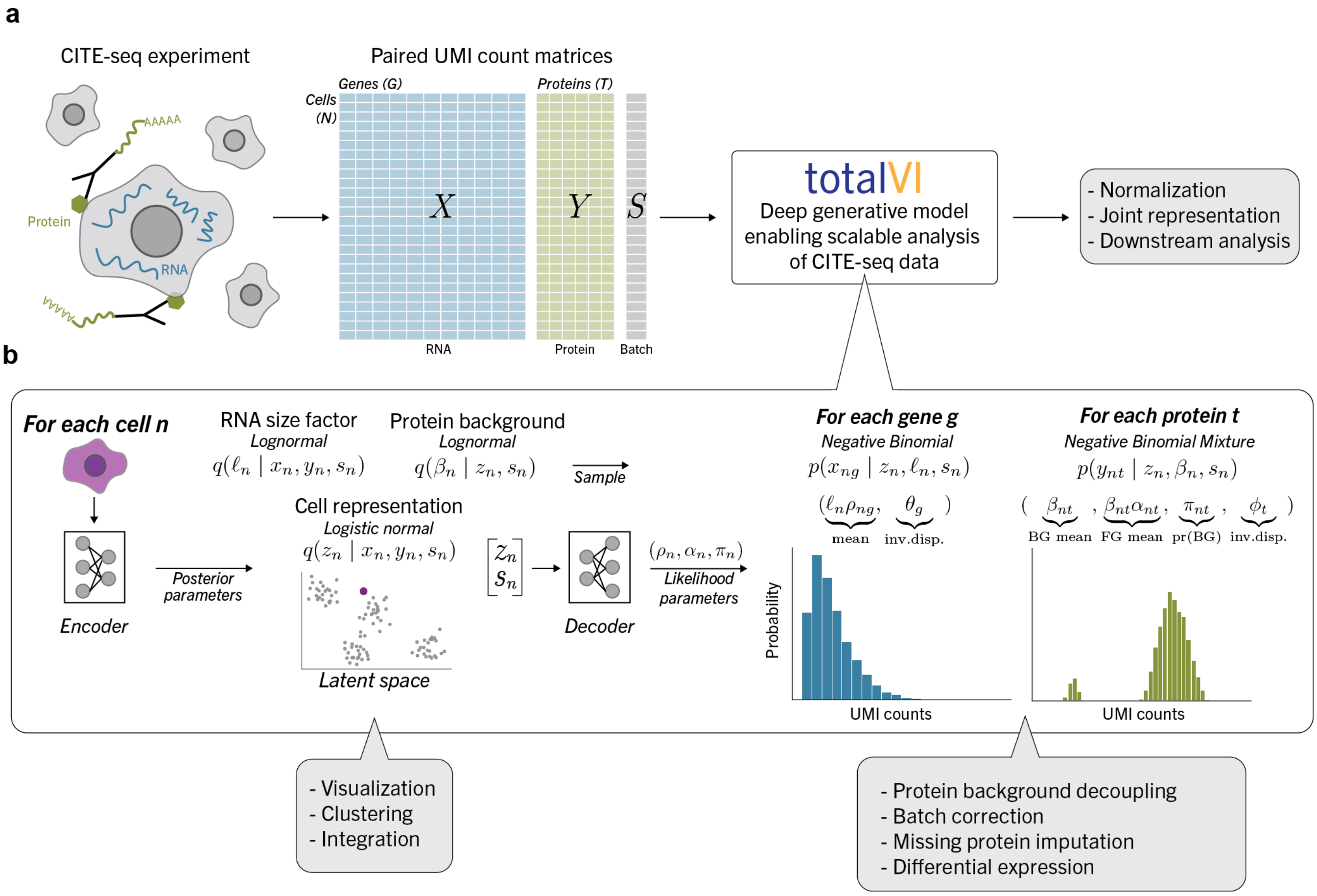
a
CITE-seq experiment
Protein
RNA
AAAAA
Paired UMI count matrices
Genes (G)
Proteins (T)
Cells (N)
X
Y
S
RNA
Protein
Batch
totalVI
Deep generative model enabling scalable analysis of CITE-seq data
- Normalization
- Joint representation
- Downstream analysis
b
For each cell n
Encoder
Posterior parameters
RNA size factor
Lognormal
$q(\ell_n \mid x_n, y_n, s_n)$
Protein background
Lognormal
$q(\beta_n \mid z_n, s_n)$
Sample
Cell representation
Logistic normal
$q(z_n \mid x_n, y_n, s_n)$
Latent space
$\begin{bmatrix} z_n \\ s_n \end{bmatrix}$
Decoder
$(\rho_n, \alpha_n, \pi_n)$
Likelihood parameters
For each gene g
Negative Binomial
$p(x_{ng} \mid z_n, \ell_n, s_n)$
$(\underbrace{\ell_n \rho_{ng}}_{\text{mean}}, \underbrace{\theta_g}_{\text{inv.disp.}})$
Probability
UMI counts
For each protein t
Negative Binomial Mixture
$p(y_{nt} \mid z_n, \beta_n, s_n)$
$(\underbrace{\beta_{nt}}_{\text{BG mean}}, \underbrace{\beta_{nt}\alpha_{nt}}_{\text{FG mean}}, \underbrace{\pi_{nt}}_{\text{pr(BG)}}, \underbrace{\phi_t}_{\text{inv.disp.}})$
UMI counts
- Visualization
- Clustering
- Integration
- Protein background decoupling
- Batch Correction
- Missing protein imputation
- Differential expression

Figure 1: Schematic of a CITE-seq data analysis pipeline with totalVI. a, A CITE-seq experiment simultaneously measures RNA and surface proteins in single cells, producing paired count matrices for RNA and proteins. These matrices, along with an optional matrix containing sample-level categorical covariates (batch), are the input to totalVI, which concomitantly normalizes the data and learns a joint representation of the data that is suitable for downstream analysis tasks. **b**, Schematic of totalVI model. The RNA counts, protein counts, and batch for each cell n are jointly transformed by an encoder neural network into the parameters of the posterior distributions for z_n, a low-dimensional representation of cell state, β_n, the protein background mean indexed by protein, and ℓ_n, an RNA size factor. The space of all joint cell representations (latent space) is independent of the batch and is used for dataset integration and cell stratification tasks. Next, a decoder neural network maps samples from the posterior distribution of z_n, along with the batch, s_n, to parameters of a negative binomial distribution for each gene and a negative binomial mixture for each protein, which contains a foreground (FG) and background (BG) component (Methods 4). These parameters are used for downstream analysis tasks.

maps z_n and s_n to ρ_{ng}, which represents the normalized gene expression. This modeling strategy for the RNA data is closely related to previous work in this domain (e.g., scVI [13], ZINB-WAVE [16], and DCA [24]).

For the protein data, totalVI posits that the number of molecules for each protein t and cell n, y_{nt} follows a two-component negative binomial mixture distribution. The mixture accounts for the fact that the observed protein counts may correspond to either real protein abundance on the cell surface, or to technical artifacts like ambient antibodies and non-specific binding. The mean of this mixture depends on z_n (similarly to RNA) and on another latent variable, β_{nt}, which follows a lognormal distribution and describes a baseline level of protein counts due to background. An additional neural network maps z_n and s_n to the protein likelihood parameters α_n and π_n, which represent an offset for the protein foreground mean, and the probability that the observed counts were generated from

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

the background component, respectively.

totalVI optimizes the parameters of both of its components simultaneously using the variational autoencoder (VAE) framework [25]. In the first component, "encoder" neural networks are trained to map the raw observations to the parameters of the approximate posterior distributions of the latent variables z_n, ℓ_n, and β_n. In the second component, the previously described "decoder" neural networks are trained to map z_n and s_n to the RNA and protein likelihood parameters ρ_n, α_n, and π_n. The inverse dispersion parameters for genes θ_g and proteins ϕ_t are model parameters learned during inference. Accordingly, totalVI uses highly efficient techniques for stochastic optimization of VAEs, which makes it appropriate for the scale of CITE-seq data.

The latent variable z_n in the first component of totalVI is particularly important for downstream analysis. As z_n is linked to the mean of the likelihood for each protein and each gene in each cell by a neural network, its approximate posterior, $q(z_n \mid x_n, y_n, s_n)$, captures information contained in both RNA and protein measurements. This distribution should also be robust to the nuisance factors accounted for, such as negative binomial noise and batch effects. We used the mean of z_n under the approximate posterior as an integrated, joint view of cell state, which we then used as input to clustering and visualization algorithms. Additionally, we modeled z_n with a logistic normal distribution, meaning each z_n resides in a convex polytope amenable to archetypal analysis [26, 27]. Such an analysis connects each latent dimension to the expression of genes and proteins and aids with the interpretation of the model.

Other downstream tasks specific to genes and proteins (collectively referred to as features), are linked to the likelihood parameters from the second component of totalVI. In the following, we demonstrate how to use this part of the model in a range of tasks such as differential expression testing, missing protein imputation, and measurement denoising. A detailed specification of the model, along with further mathematical descriptions of the quantities used in downstream tasks is in Methods 4.

2.2 CITE-seq profiling of murine spleen and lymph nodes

We conducted a series of CITE-seq experiments that were designed to test the performance of totalVI on a variety of tasks while handling issues related to large protein panels that can vary between different batches. As a case study, we profiled murine spleen and lymph nodes, which contain heterogeneous immune cell populations that are well-characterized by surface protein markers. The CITE-seq experiments were executed on the 10x Genomics Chromium microfluidic platform using BioLegend TotalSeq-A antibodies.

In these experiments, cells were collected from two wild-type mice that were processed in two different experimental runs to serve as biological replicates (Methods 4.4). For each experimental run (completed on separate days), we stained cells from one mouse with two different panels of barcoded antibodies: the first contained 111 antibodies and the second contained 208 antibodies, of which the 111 antibodies were a subset (Supplementary Data: Antibodies). Spleen and lymph node cells were stained separately with the same antibody panels plus an additional hashtag antibody such that the tissues could be demultiplexed [28]. Cells stained with the same panel were pooled in the same 10x lane, resulting in four lanes (two per run).

We refer to these four spleen/lymph node datasets by their protein panel and experimental day: SLN111-D1, SLN208-D1, SLN111-D2, SLN208-D2. We refer to the combination of the four as SLN-all (experimental design in Table 1). After pre-processing and filtering, these datasets contained a total of 32,648 cells with approximately 3,300 UMIs per cell for RNA and 2,800 UMIs per cell for proteins (Methods 4.6).

2.3 totalVI is scalable and fits CITE-seq data well

The usefulness of probabilistic models like totalVI depends on how well they fit the observed data. Furthermore, they should generalize to unobserved data (i.e., not overfit) and scale to a realistic range of input sizes. Here we verify that totalVI satisfies these prerequisites.

We first estimated how well totalVI fits CITE-seq data that was available to it during training using posterior predictive checks (PPC; [14, 29]). To conduct PPCs, we generated replicated datasets (i.e., posterior predictive samples) by sampling from the fitted model (Methods 4.7). One replicated dataset is constructed by passing each observed cell n through the encoder part of the model,

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

sampling from the posterior approximation of its latent variables (e.g., z_n), and then sampling from the conditional distributions for each gene and protein given the latent variables (Figure 1b). As totalVI directly models the count data, the replicated datasets generated with this procedure should maintain the properties of the observed data. To that end, we assessed how well totalVI preserves the mean-variance relationship of the data by measuring the similarity between the coefficient of variation (CV) per gene and protein on the replicated and the raw data.

We then evaluated how well totalVI generalizes to data that was not available during training (i.e., a set of held-out cells). To do so, we generated replicated datasets conditioned on the held-out cells, and computed two somewhat opposing metrics of predictive performance. First, we assessed how well the average replicated data set matched the observed held-out data via mean absolute error. Second, we quantified how well the interval of values from replicated data sets covered the observed held-out data values (calibration error; [30]). We also computed the held-out predictive log-likelihood of the data. All of these held-out metrics were computed separately for genes and proteins.

As a baseline, we compared totalVI to factor analysis (FA), which can be viewed as a simplified version of totalVI. For instance, in FA, the mapping between the latent space and data space is specified by a linear function and the conditional distribution for each feature is Gaussian. As an additional control, we compared to scVI [13], which was restricted to the RNA part of the data. We expected the goodness of fit for totalVI and scVI on just the RNA part of the data to be comparable, as they share a similar generative model and neural network architecture.

Our evaluation relied on fitting the models to several CITE-seq datasets, spanning different species and tissues, including peripheral blood mononuclear cells (PBMC10k; [31]) and mucosa-associated lymphoid tissue (MALT; [32]) from humans, and our novel murine spleen and lymph node data (SLN111-D1). Across all datasets, totalVI performed best in terms of preserving the mean-variance relationship of genes and proteins in the raw data (Supplementary Figure S1a). On the held-out protein data, totalVI outperformed FA in both the mean absolute error and calibration error metrics. On the held-out RNA data, totalVI and scVI were largely comparable and outperformed FA (Supplementary Figure S1b-d).

To assess the scalability of totalVI, we concatenated all of our spleen and lymph node data (SLN-all) and recorded the training time for different sizes of subsets of this data. totalVI and scVI (which processes only RNA data) had similar dependence between run time and input size (Supplementary Figure S1e). Furthermore, we observed that totalVI can readily handle large data sets, for instance, processing the complete set of approximately 33,000 cells with over 4,100 features in less than an hour.

Overall, these results demonstrate that totalVI is a scalable probabilistic model that provides an accurate representation of CITE-seq data. With these prerequisites satisfied, we turned to benchmarking particular aspects of the model as they relate to common single-cell analysis tasks.

2.4 totalVI identifies and corrects for protein background

To analyze protein data in an accurate and quantitative manner, it is necessary to distinguish between true biological signal and technical bias in the protein measurement. Background is a type of technical bias that has previously been reported for antibody-based measurements and has been observed in both flow cytometry [33] and sequencing-based [8, 9] studies. In CITE-seq data, protein counts contain non-negligible background, the extent of which can be observed in the detection of counts for nearly every protein in all cells. This background arises experimentally from a combination of unbound ambient antibodies, which can be detected in empty droplets, and non-specific antibody binding, which can be detected in cells with no expected expression of a protein, such as CD19 in T cells (Methods 4.8, Supplementary Figure S2a-c, g). Recent methods have described background from ambient RNA [34, 35], but the presence of background is more pronounced in protein measurements (Supplementary Figure S2d-f). Failure to account for this protein background could lead to erroneous conclusions in downstream analyses. For instance, differential expression tests might report false positives driven by variation in background rather than by the biological signal of interest, as well as false negatives if high levels of background obscure low levels of signal.

Accurate identification of protein foreground and background is a prerequisite for denoising the protein measurements. Previous studies of CITE-seq data derived a single decision rule for every protein, specifying the minimal number of counts required to be considered foreground. In one study,

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

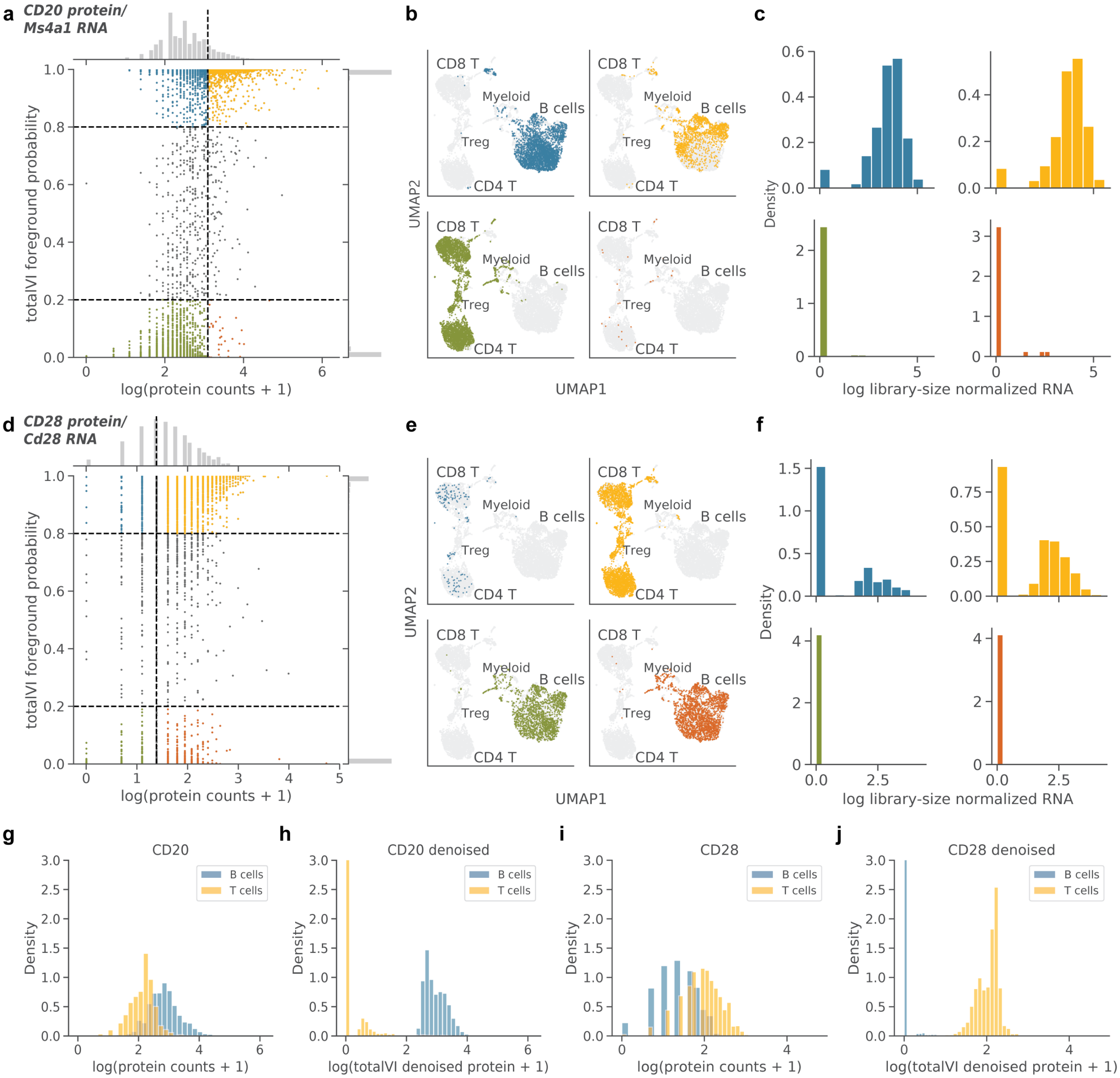

Figure 2: totalVI identifies and corrects for protein background. totalVI was applied to the SLN111-D1 dataset. **a-c**, CD20 protein (encoded by Ms4a1 RNA). **(a)** totalVI foreground probability vs log(protein counts + 1). Vertical line denotes protein foreground/background cutoff determined by a GMM. Horizontal lines denote totalVI foreground probability of 0.2 and 0.8. Cells with foreground probability greater than 0.8 or less than 0.2 are colored by quadrant, while the remaining cells are gray. **(b)** UMAP plots of the totalVI latent space. Each quadrant contains cells from the corresponding quadrant of (a) in color with the remaining cells in gray. **(c)** RNA expression (log library-size normalized; Methods 4.8) for cells colored in (a). **d-f**, Same as (a-c), but for CD28 protein (encoded by Cd28 RNA). **g, h**, Distributions of log(protein counts + 1) **(g)** and log(totalVI denoised protein + 1) **(h)** for CD20 protein in B and T cells. y-axis is truncated at 3. **i, j**, Same as (g, h), but for CD28 protein.

the boundary was based on spiked-in negative control cells from another species, relying on the assumption that cross-reactivity of antibodies between species is minimal [8]. Another study fit a

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Gaussian mixture model (GMM) to distinguish a background and foreground component for each protein [36]. Using the same boundary for all cells, however, relies on the assumptions that all cells are subject to a similar background distribution of the protein in question and, for the commonly applied case of a two-component GMM, that the foreground component is comparable across cell types.

Since these assumptions might not hold in all cases, totalVI instead models protein background as cell- and protein-specific. In totalVI, the estimate of protein background accounts not only for the observed levels of a given protein in the cell in question, but also for the overall transcriptomic and proteomic profile of that cell. To do this, totalVI models each protein measurement as a mixture of a foreground and background component that depends on the cell's representation in the latent space (Methods 4.1). The mixture is weighted by the parameter π, which can be interpreted as the probability that the counts of a protein in a given cell came from the background component (Figure 1b, Methods 4.8).

As a way of evaluating totalVI's ability to quantitatively identify protein background using π, we made use of common marker proteins, whose counts are expected to come from the foreground component in some cell types and from the background component in others. For example, a high foreground probability for CD4 could be used as a positive predictor of CD4 T cells. We tested how well the foreground probability estimates $(1 - \pi)$ of each marker protein in the SLN111-D1 dataset performed at classifying major cell types by computing the area under the receiver operating characteristic curve (ROC AUC), using manual cell type annotations as the true labels. As a baseline for comparison, we used the assignment probabilities from a two-component Gaussian mixture model (GMM) fit on all cells for each log-transformed protein (Methods 4.8).

For nine out of eleven known marker proteins, both totalVI and the GMM performed well at classifying cell types by protein foreground probability (ROC AUC > 0.97; Supplementary Figure S3a). For these marker proteins, distinguishing foreground and background components was relatively straightforward, and foreground and background levels appeared to be similar across all cells. The B cell marker CD19 was one example where the distributions of foreground and background protein expression were easily separated (Supplementary Figures S2a and S4a-d).

Other proteins, however, were not as simple, and had highly overlapping distributions of foreground and background counts (Supplementary Figure S2b, c). For two of these more challenging proteins, namely, the B cell marker CD20 and the T cell marker CD28, totalVI noticeably outperformed the GMM (Supplementary Figure S3b). These two proteins are illustrative of two types of errors that can be made when using a single decision boundary on all cells to distinguish protein foreground from background.

In the case of CD20 (encoded by Ms4a1 RNA), the GMM set a high cutoff on protein counts that resulted in numerous false negatives (Methods 4.8). Among the cells below the cutoff (considered CD20 negative by the GMM), totalVI distinguished cells with high and low foreground probability (Figure 2a). The cells with high foreground probability clustered with B cells in the totalVI latent space and had high Ms4a1 expression, confirming their identity as B cells. On the other hand, cells with similarly low protein expression but with low probability foreground clustered with T cells and had no Ms4a1 expression (Figure 2a-c).

In the case of CD28, the GMM set a low cutoff on protein counts that resulted in numerous false positives. Among the cells above the GMM cutoff, totalVI distinguished the cells with high foreground probability (primarily T cells) from the cells with low foreground probability (primarily B cells) (Figure 2d-f). Across all proteins, the totalVI foreground probability tended to fall near zero or one (Supplementary Figure S4e).

Despite using a two-component mixture, totalVI can also decouple background and foreground for proteins that have greater than two modes of expression globally. For example, totalVI correctly decoupled background and foreground for CD4 in peripheral blood mononuclear cells, which globally follows a trimodal distribution (Methods 4.8, Supplementary Figure S5a, b). Through the above examples, we see that by leveraging information from all cells, genes, and proteins, totalVI is capable of identifying protein foreground and background in a way that more accurately reflects cell state than a single decision boundary on protein expression.

To illustrate background identification, we demonstrated that decoupling foreground from background can be used to "classify" a cell as expressing or not expressing a certain protein. For downstream

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

analysis, totalVI uses π in a more quantitative manner to remove protein background. Specifically, totalVI can denoise the protein data by setting the background component to zero, while also accounting for the measurement uncertainty of the foreground component. This effectively subtracts the expected amount of background from the overall expected expression (Methods 4.3 and 4.8, Figure 2g-j, Supplementary Figure S5f, g).

Statistics computed on the expectation of denoised values may be biased due to spurious relationships between features that arise in the denoising process [37]. We used the expectation of denoised values for visualization (Supplementary Figure S5c-e). However, to address this concern for statistical analyses, the totalVI differential expression test uses distributions over the denoised values as opposed to testing directly on a denoised data matrix; we further describe this procedure below. For other analyses focused on the relationships between features, we developed a novel sampling method for denoised data that controls for nuisance variation while avoiding denoising-induced artifacts (Methods 4.3). We applied this method to construct feature-feature correlation matrices and found that totalVI preserved the independence of negative control genes, while the naive calculation of correlations on expected denoised values generated false positive correlations (Methods 4.9, Supplementary Figure S6a, b, d, e). These results lend confidence that downstream analysis with totalVI is not subject to spurious feature relationships that can arise from data denoising. When observing the correlations between proteins and their encoding RNA, we found that totalVI correlations were higher in magnitude than raw correlations across the majority of RNA-protein pairs (Supplementary Figure S6c, f). It is worth noting that the totalVI model has no explicit information about the relationship between any RNA-protein pairs, such that any correlation learned by the model is not predetermined by known RNA-protein relationships. In the sections below, we demonstrate how denoising protein expression within totalVI results in more accurate downstream analysis.

2.5 Integration of multiple datasets

As the number of CITE-seq studies grows, we anticipate increased interest in performing integrated analyses of datasets from different labs, experiments, and technologies; however, these categorical covariates often confound the shared biological signal between datasets. In the case of scRNA-seq data, methods like Seurat v3 [38] and Scanorama [39] were designed to correct expression values for these so-called batch effects and have demonstrated state-of-the-art results. These methods can also be extended to integrate CITE-seq datasets, but their application is limited. First, their corrected values are not designed to be used for differential expression testing or other related downstream tasks, since the uncertainty of the correction is not considered. Second, they require each dataset to have the same features, which may not be the case due to differences in antibody panels between datasets.

Integration is built into the totalVI model via an assumption of independence between the latent space and the batch. Consequently, the batch is treated as a covariate in the totalVI generative model, allowing for a statistically robust integration procedure. Furthermore, totalVI can handle datasets with different antibody panels due to a novel modification in which datasets with different proteins are processed using a fixed neural network architecture designed for the union of the proteins [40]. totalVI can also impute the expression values of missing proteins by making out-of-sample predictions, which incorporates the information learned from those proteins in the batches in which they were observed (Methods 4.3, [41]).

We benchmarked totalVI against state-of-the-art methods using general metrics designed to test how well datasets are mixed and how well the original structure of each dataset is maintained. We used these metrics to demonstrate that totalVI can effectively integrate datasets with equal and unequal protein panels. Finally, we benchmarked totalVI's accuracy of predicting protein expression in cases where measurements are available in only one of the datasets.

We considered three metrics to quantify integration performance. First, the latent mixing metric quantifies how well the different batches mix in the low-dimensional latent space by considering the frequency of observed batches within local cell neighborhoods [42]. The second performance criterion, the measurement mixing metric, describes how well the different batches mix in the high-dimensional data space. This test uses the Mann-Whitney U statistic to evaluate the extent to which the expression of a given feature is systematically higher in one of the batches post-integration. Finally, as the mixing metrics could be trivially optimized by randomizing the data, we also tested how well the patterns of feature expression in every individual dataset are preserved in the integrated dataset.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

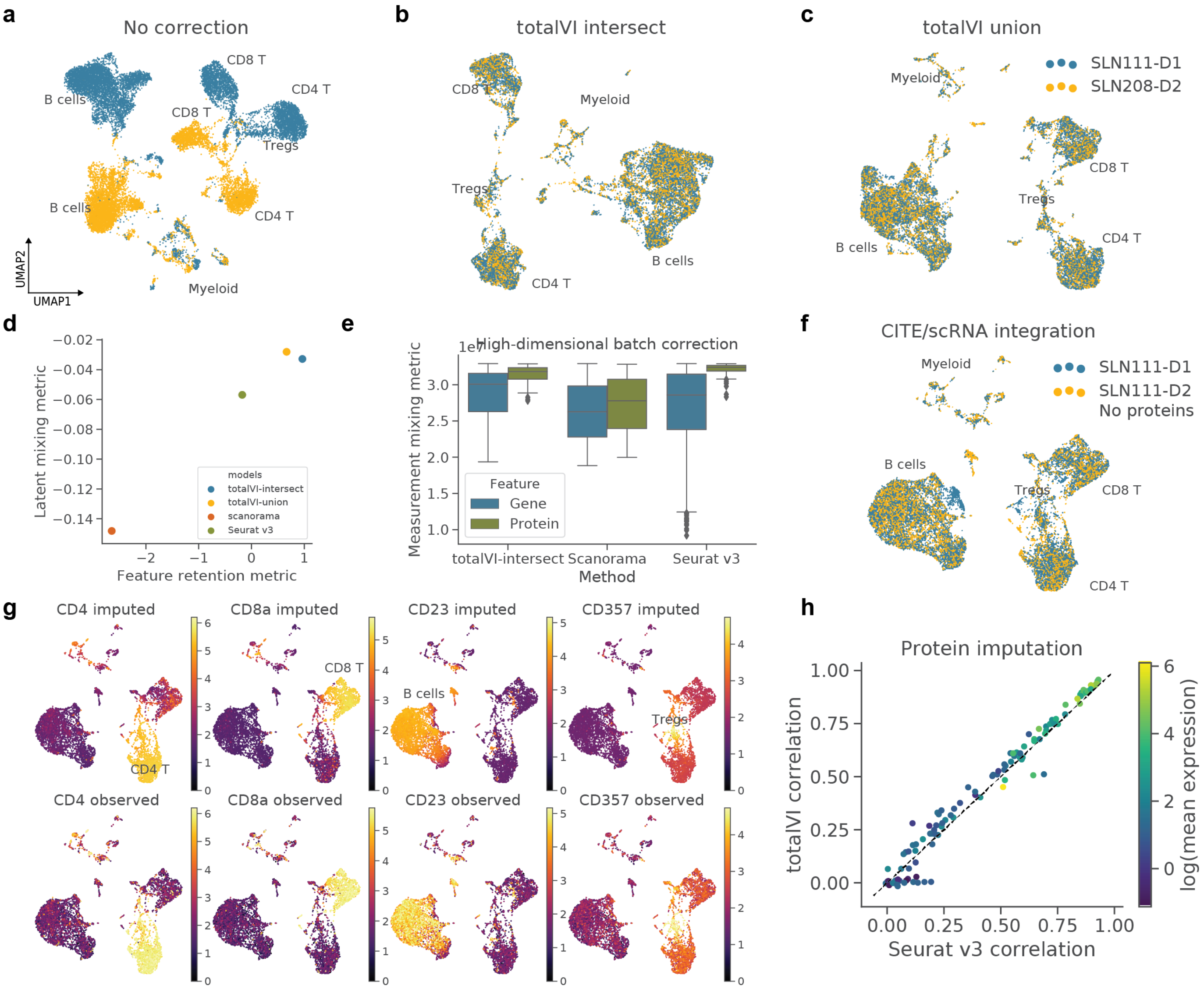

Figure 3: Benchmarking of integration methods for CITE-seq data. **a-c**, UMAP plots of SLN111-D1 and SLN208-D2 with no integration (PCA of paired data with intersection of protein panels), and after integration with totalVI-intersect, in which the protein panels were intersected, and totalVI-union, in which the unequal protein panels were preserved, colored by dataset. **d, e**, Performance of integration methods based on three metrics: **(d)** latent mixing metric, feature retention metric and **(e)** measurement mixing metric (higher values are better for each; Methods 4.10). **f**, UMAP plot of SLN111-D1 integrated with SLN111-D2 (proteins held out) by totalVI. **g**, UMAP plots colored by totalVI imputed and observed protein expression (log scale) of key cell type markers (range 0-99th percentile of held-out values for each protein). **h**, Pearson correlation of imputed versus observed protein expression (log scale) for totalVI-union and Seurat v3 colored by average expression of the protein.

In particular, we used the idea of autocorrelation, which measures for each feature the extent to which cells that are near each other in latent space have a similar expression level [43]. The feature retention metric summarizes the difference in the autocorrelation of features computed using the latent space of each individual dataset versus the integrated latent space. Thus, it tests whether the integrated latent space captures similar genes and proteins as the latent space of each dataset in isolation (Methods 4.10).

We demonstrated totalVI's integration capabilities on two of the spleen and lymph node datasets

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

(SLN111-D1, SLN208-D2), which were generated with different panels containing either 111 or 208 antibodies (111 is a subset of the 208). For these datasets, a linear dimensionality reduction with principal components analysis (PCA) revealed batch as a major source of variation (Figure 3a). We compared totalVI to Seurat v3 [38] and Scanorama [39], which, like totalVI, produce both batch corrected expression values and an integrated latent space. We benchmarked two versions of totalVI – totalVI-intersect, in which protein features were intersected, and totalVI-union, in which the union of protein features was used as input.

For each of these methods, we computed UMAP embeddings [44] of their latent spaces (Figure 3b, c), as well as the three evaluation metrics. Generally, we found that after integration, cells of similar types were co-located in the latent space, as evidenced by the shared expression of key marker proteins like CD4, CD8a, and CD19 (Supplementary Figure S7). In the latent mixing and feature retention metrics, totalVI outperformed the other methods, while comparing favorably in the measurement mixing metric (Figure 3d, e). totalVI-union and totalVI-intersect performed similarly, indicating that the presence of missing data did not diminish totalVI's integration capabilities. We also repeated this analysis on two public datasets of PBMCs (PBMC10k [31], PBMC5k [45]) and observed similar performance for totalVI (Supplementary Figure S8a-f).

Since totalVI-union can integrate datasets with different protein panels, we reasoned it would be feasible to integrate a CITE-seq dataset with a standard scRNA-seq dataset that has not measured proteins and impute the missing protein measurements. We assessed this by integrating SLN111-D1 and SLN111-D2, where we held out the proteins of SLN111-D2. We first observed that totalVI can learn a biologically meaningful integrated latent representation despite the large amount of missing data (Figure 3f). Indeed, the localization of observed protein expression in the latent space revealed the same broad immune cell types like B cells, CD4 T cells, CD8 T cells, Tregs, and myeloid cells. Next, we decoded the latent representations of cells from SLN111-D2 conditioned on them being in SLN111-D1, where protein data was observed, and reported the mean of the negative binomial mixture for each protein as the imputed protein values (Methods 4.3). For key cell type marker proteins, totalVI-imputed proteins shared similar patterns of expression as the held-out observed proteins (Figure 3g).

To further quantify imputation accuracy, we computed the Pearson correlation between imputed and observed protein values on the log scale. We did not correct for background in this analysis since the comparison is to the observed data. We compared totalVI to Seurat v3, which imputes protein values based on smoothing of protein values from mutual nearest RNA neighbors. totalVI had higher correlation than Seurat v3 for the majority of the proteins (Figure 3h). As expected, imputation was inaccurate in both methods for proteins with low counts, which may only have foreground signal in few cells, or were poorly detected. We observed similar results when applying this task to PBMCs (Supplementary Figure S8g, h). Taken together, these results suggest that imputed proteins may be used as a proxy for real protein measurements in other downstream tasks.

2.6 Differential expression

totalVI can leverage its estimates of uncertainty from a single model fit to detect differentially expressed features between sets of cells while controlling for noise and other modeled technical biases. In a test between two sets of cells, totalVI estimates the posterior odds that any feature is differentially expressed (using Bayes factors [13, 46, 47]; Methods 4.3). Here, the Bayesian equivalent of a null hypothesis for a particular feature is that the log fold change (LFC) of expression between the two sets is contained within a small interval centered at zero. Likewise, the probability that the LFC is outside this interval is considered the *probability of differential expression* (DE). The Bayes factor thus quantifies the degree to which the data support the hypothesis that a feature is differentially expressed versus the hypothesis that it is not differentially expressed. A distribution over the LFC is estimated using posterior samples of z_n and the subsequent data likelihood parameters (Figure 1b), adjusted for sequencing depth (RNA), background signal (protein), and batch effects (both). We used the median of this LFC distribution as an estimate of effect size.

To evaluate totalVI as a framework for DE analysis in the common scenario of multiple experiments, we integrated all four spleen and lymph node datasets (SLN-all; totalVI-intersect). totalVI provided a descriptive representation of this data, as clusters of cells in the joint latent space corresponded to immune cell types or states. Indeed, we found consistent differences in the expression of known immune cell markers, thus allowing us to manually annotate each cluster with cell type labels

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

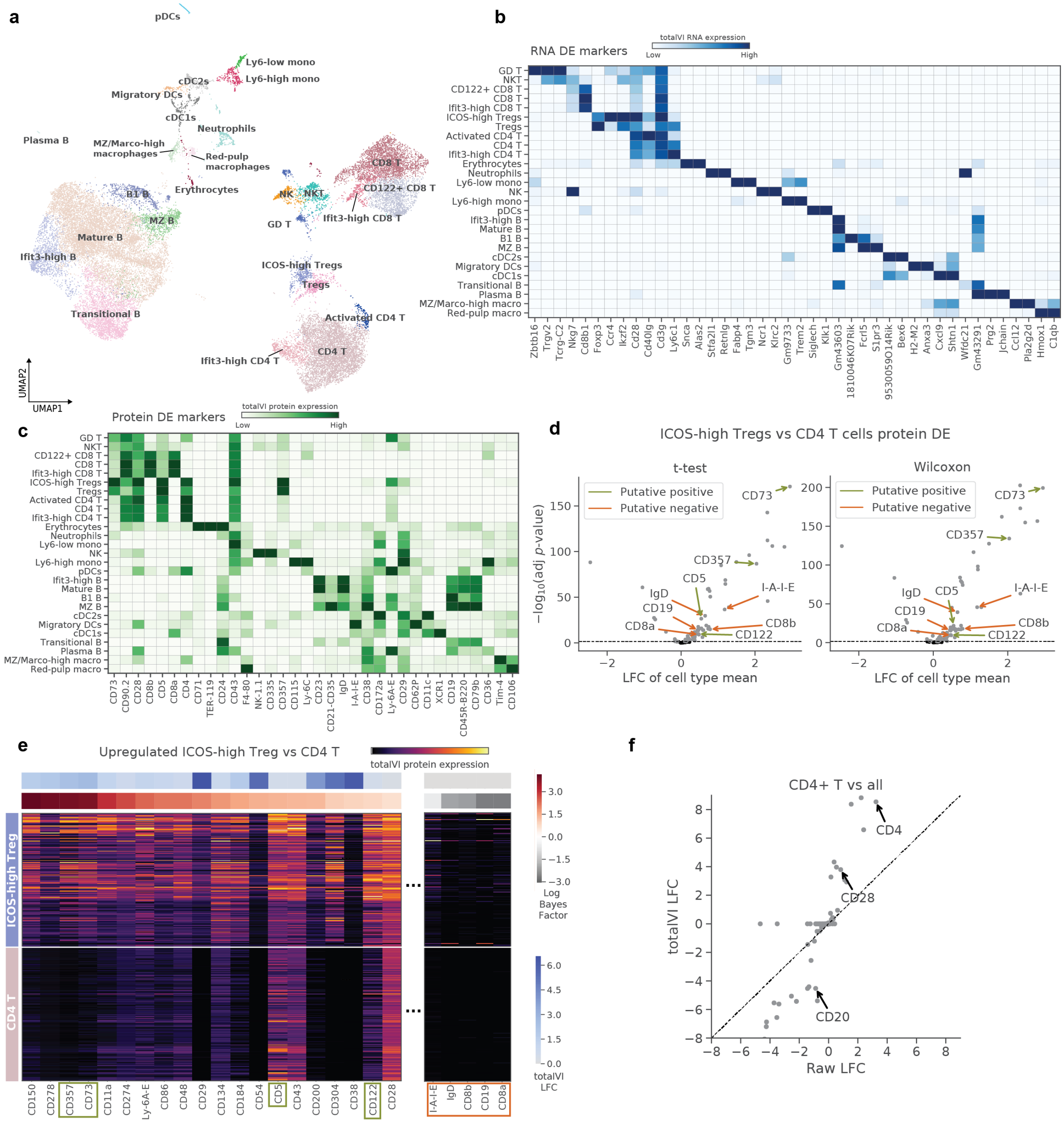

Figure 4: totalVI identifies differentially expressed genes and proteins. totalVI intersect was applied to the SLN-all dataset. **a**, UMAP plot of SLN-all, after clustering and annotating the data (Methods 4.11). **a, b**, Heatmap of markers derived from one-vs-all tests for **(a)** RNA and **(b)** proteins. For each cell type, we display the top three protein markers and top two RNA markers in terms of LFC. **d**, Volcano plot of protein differential expression test between ICOS-high Tregs and CD4 T cells for a Welch's t-test and Wilcoxon rank-sum test. Putative positives and negatives are denoted by green and orange arrows, respectively. Significant proteins are colored in grey, all others are in black. **e**, totalVI protein expression for proteins (columns) upregulated in ICOS-high Tregs versus CD4 T cells. Cells (rows) are ordered by cluster, and subsampled to be equal in number per cluster. Columns are normalized in the range [0, 1]. The left section in the heatmap contains all the proteins called differentially expressed by totalVI with a positive log fold change. Proteins are sorted by Bayes factor (significance). The rightmost section contains the putative negatives, which are not called differentially expressed by totalVI. **f**, Comparison of log fold changes estimated by totalVI and observed in the raw data from a one-vs-all test of CD4 T cells.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

(Figure 4a, Supplementary Figure S9, Methods 4.11). These annotations were consistent with the latent space derived with totalVI-union (Supplementary Figure S10), and we used them throughout this section.

Beyond markers used for annotation, we found that a totalVI one-vs-all DE test (in which one cell type is compared to all others) identified many additional features as differentially expressed by one or more subsets of cells (Methods 4.12, Figure 4b, c). For example, totalVI identified the gene Klrc2 as differentially expressed in both natural killer (NK) cells and gamma/delta T cells, which has previously been shown to be upregluated in these populations relative to alpha/beta T cells [48]. On the protein side, totalVI identified CD335 (NKp46) as among the top protein markers for NK cells, which is a canonical marker used for sorting [49], and CD43, which is associated with the development of NK cells [50].

Overall, the Bayes factors inferred by totalVI for the RNA data were highly correlated with those produced by scVI (Supplementary Figure S11a), which has been independently evaluated [47]; therefore, we focused on evaluating the protein DE test. For the protein DE test, we focused on testing for accurate detection of true positive and negative cases of DE and reproducibility across datasets. Throughout, we compared totalVI to two methods – a Welch's t-test applied to the log-transformed protein counts and a Wilcoxon rank-sum test. These general methods are commonly used for scRNA-seq data, so we used them as baselines for protein DE testing (Methods 4.12).

We first evaluated the extent of false positives using isotype control antibodies. As isotype controls lack target specificity, differences in the abundance of isotype controls between cell types may stem from differences in background or other technical sources of variation. We applied totalVI to the SLN208-D1 dataset, which contained nine isotype controls, and performed a one-vs-all DE test. In all but one of the comparisons in the one-vs-all test, totalVI called zero of the isotype controls as differentially expressed, outperforming the baseline methods. totalVI also outperformed a version of itself (totalVI-wBG) in which the protein background component is not subtracted (Supplementary Figure S11b).

To gain further insight into the extent of false positive and false negative DE calls, we compared ICOS-high regulatory T cells (ICOS-high Tregs) and conventional CD4 T cells from SLN-all in a one-vs-one DE test. This test is challenging because these two cell types share many of the same upregulated and downregulated features when compared with other immune cell types. Our analysis was based on a list of putative positives and negative surface proteins curated from previous studies that used flow cytometry. In this test, we expected CD73, CD357 (GITR), CD122, and CD5 to be upregulated (positives) in ICOS-high Tregs relative to conventional CD4 T cells [51–54].

We found that all three methods identified these positives as significantly upregulated; however, the two baseline methods also incorrectly called all putative negatives as upregulated, which included I-A/I-E (MHC II), IgD, CD19, CD8b, and CD8a, which have no expected expression in either of these cell types (Figure 4d). Globally, the two baseline methods called the majority of the proteins in the panel (78/111 proteins in both cases) as differentially expressed, many of which are likely the result of differences in background. Filtering proteins by the observed LFC in the baseline methods may reduce false positives in these cases, but the increased accuracy would still be limited (e.g., CD5 and IgD had a similar LFC and therefore could not be distinguished; Figure 4d). The totalVI test, on the other hand, correctly classified the putative negatives and putative positives in our curated list (Figure 4e), while calling 28 proteins differentially expressed in total. To further support the utility of removing background from the protein data, we performed this test using totalVI-wBG, which improved upon the baseline methods, but also falsely called I-A/I-E (MHC II) and CD8b as positives (Supplementary Figure S12a).

To test for reproducibility across biological replicates, we applied each method separately to the SLN111-D1 and SLN111-D2 datasets in a one-vs-all DE test. Comparing the results obtained from each dataset (Bayes factors for totalVI and adjusted p-values for the benchmark methods) by Spearman correlation, we found that totalVI had the highest consistency between replicates (Spearman's $\rho = 0.91$; Supplementary Figure S11c-e). We also tested for reproducibility across experimental designs: one in which the two CITE-seq datasets (SLN111-D1, SLN111-D2) had equal protein panels, and another in which proteins were measured in only one of the datasets. We used totalVI to integrate the two batches in each scenario and conducted one-vs-all DE tests. The resulting Bayes factors were reproducible between the two scenarios (Spearman's $\rho = 0.86$; Supplementary Figure S11f).

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Finally, the LFC estimates calculated by totalVI also better captured the underlying biological signal. For example, in a test of CD4 T cells vs all from SLN-all, the canonical marker CD4 had a higher LFC than in the raw data (Figure 4f). Additional markers like CD28 (T cell marker) and CD20 (B cell marker), which we previously highlighted as having highly overlapping foreground and background components, had respectively higher and lower LFCs compared to LFCs derived from the raw data. This increase in contrast is driven by totalVI's ability to probabilistically remove the background.

2.7 Interpretation of totalVI latent space

Deep-learning-based methods for dimensionality reduction tend to rely on "black-box" models, making it difficult to interpret the coordinates of their inferred low-dimensional latent spaces. For example, for autoencoder-based single-cell methods like scVI or DCA [13, 24], there is no straightforward way to determine which expression programs are associated with each dimension of the latent space. This is in contrast to linear methods like PCA, GLM-PCA or LDVAE [27, 55], where each latent dimension is associated with a loading vector that describes the contribution made by each feature and thus enables direct interpretation. The interpretability of linear methods, however, comes at the expense of reduced capacity to fit complex data such as that obtained by scRNA-seq [27].

In totalVI, although the relationship between the latent space and the observation space is non-linear, the model still provides a way to relate each latent dimension to the expression of individual genes and proteins. Specifically, the latent vector z_n associated with each cell n follows a logistic normal distribution, such that the values in z_n are non-negative and sum to one. Thus, each cell represents a mixture over the different vertices of a probability simplex [26, 56, 57]. Each of these vertices, commonly referred to as archetypes, corresponds to a point in the latent space of totalVI, where the value of z is 1 in one coordinate and 0 in all others. The individual cells can thus be thought of as making a tradeoff, choosing the archetypes that best describes their function. The archtypes, on the other hand, represent a summary of expression programs, the combination of which characterizes a cell.

The expression profile associated with each archetype can be obtained using the decoder network of totaVI, as it is trained to map points in latent space to the parameters that underlie the distribution of each gene and protein (Figure 1b). To explore these archetypal gene and protein expression profiles, we decoded the archetypal points from the SLN-all dataset (Supplementary Figure S13a, b). We found that some archetypal points in this dataset correspond to specific cell types, whereas others corresponded to more global sources of variation that are not constrained to one cell type (Supplementary Figure S14a).

For example, archetype 16 was associated with high protein expression of CD93 and CD24, which mark the transitional B cell subset (Supplementary Figure S14b). Conversely, archetype 7 was associated with interferon-response genes such as Ifit3 and Isg20 and reflected within cell type variability in several subsets, including CD4 and CD8 T cells, B cells, Ly6-high monocytes and neutrophils (Supplementary Figures S14c and S15). Therefore, archetypal analysis enables a data-driven alternative to clustering for characterizing the heterogeneity observed both between and within cell subsets.

Archetypal analysis can also provide insight into the inner-workings of the model. In particular, we sought to evaluate the contribution of the protein data to the model and to the inferred latent space. To this end, we computed for every archetype, which percent of its top associated features are proteins. The representation of proteins amongst the top ranking features is higher than naively expected by their share of the feature population (Supplementary Figure S13c). Furthermore, the top genes often did not encode the top proteins in each archetype. Together, these results suggest that the protein data significantly influences the locations of the cells in the latent space.

2.8 Characterization of B cell heterogeneity in the spleen and lymph nodes with RNA and proteins

We next demonstrate how a joint representation of RNA and protein can be used to characterize cell identities within a specific immune compartment and in the context of multiple samples. To this end, we used the totalVI-intersect model fit on the SLN-all dataset and focused on the B cell population

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

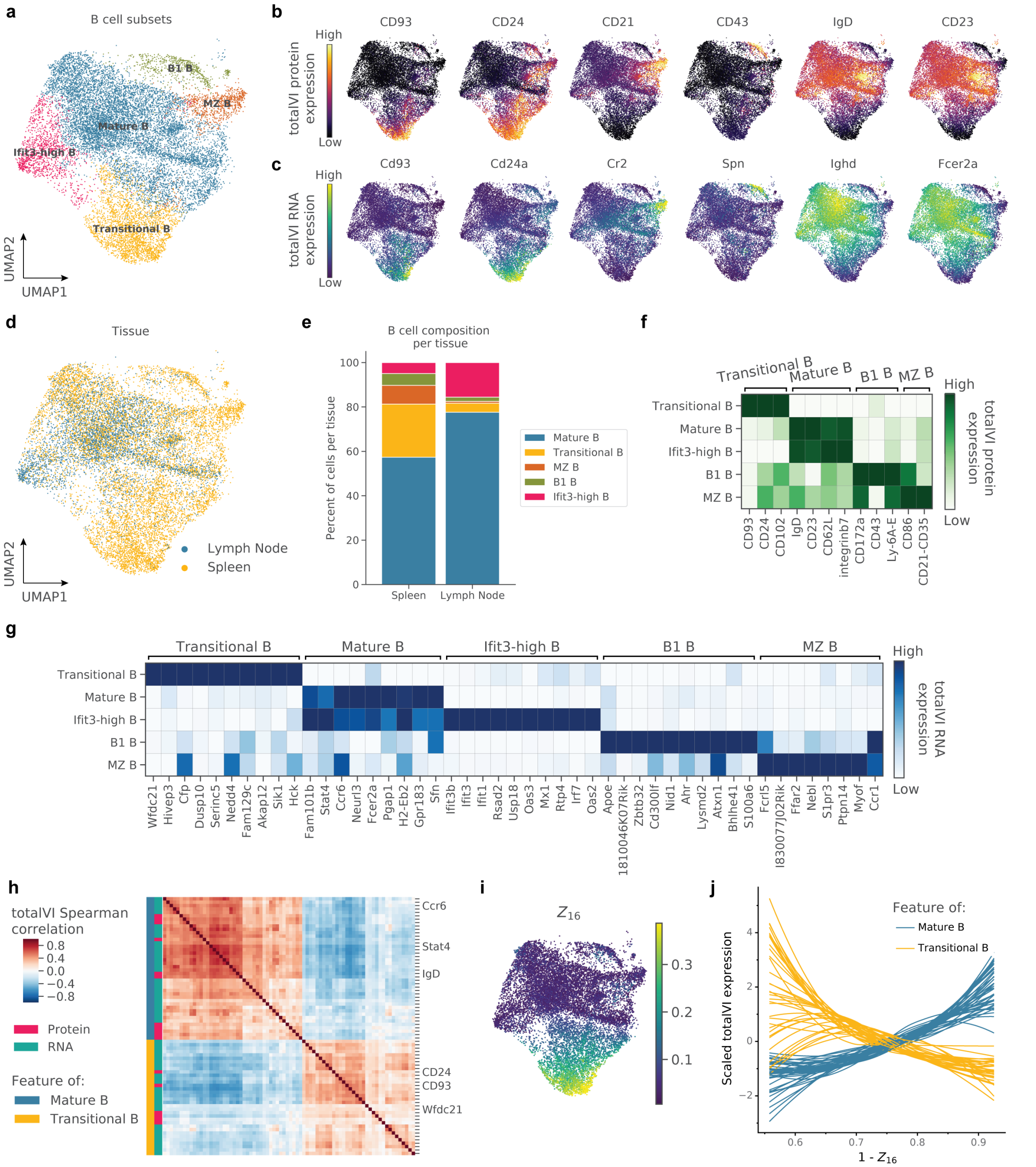

Figure 5: Characterization of B cell heterogeneity in the spleen and lymph nodes with RNA and protein. totalVI-intersect was applied to the SLN-all dataset. Data were filtered to include B cells. **a**, UMAP plot of totalVI latent space labeled by cell type. **b, c**, UMAP plots of totalVI latent space colored by **(b)** totalVI protein expression of six marker proteins and **(c)** totalVI RNA expression of the six genes that encode the corresponding proteins in (b). **d**, UMAP plot of totalVI latent space labeled by tissue. **e**, Cell type composition per tissue. **f, g**, totalVI one-vs-all differential expression test on B cell subsets filtered for significance (Methods 4.12) and sorted by the totalVI median LFC. **(f)** The top three differentially expressed proteins per subset and **(g)** the top ten differentially expressed genes per subset, arranged by the subset in which the LFC is highest. **h**, totalVI Spearman correlations in transitional B cells between RNA and proteins, which were selected as described in Methods 4.14. Features were hierarchically clustered and are labeled as either RNA or protein, and by the cell type with which the feature is associated. **i**, UMAP plot of totalVI latent space colored by Z_{16} (the totalVI latent dimension associated with transitional B cells). **j**, totalVI expression of features in (h) as a function of $(1 - Z_{16})$. Each feature was standard scaled and smoothed with a loess curve.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

(Methods 4.14, Figure 5a).

We start with characterizing cell identities using prior biological knowledge by curating a set of six surface markers that are commonly used for isolating B cell subsets (Table 4), and visualizing their expression on the UMAP representation of the totalVI latent space (Figure 5b). We found that these markers stratified the B cells into groups that were largely consistent with unsupervised clustering (Methods 4.14). RNA expression of these markers followed similar patterns to the proteins they encode (Figure 5c). This analysis identified several B cell subsets, including transitional B cells (marked by CD93 and CD24) and mature B cells (marked by IgD and CD23). Additionally, we detected clusters that represent the non-follicular subsets of B1 and marginal zone (MZ) B cells (marked by CD43 and CD21, respectively).

Comparing the composition of the different B cell subsets in the spleen and in the lymph nodes (Figure 5d), we found that these tissues contained overlapping but not identical subsets of B cells in a manner that was consistent with previous studies (Figure 5e, [58, 59]). In particular, clusters spanned the developmental range from recent bone-marrow emigrants in the splenic transitional B cell subset to a mature subset that had substantial representation in both the spleen and lymph nodes. As expected, the B1 and MZ B cell subsets were found primarily in the spleen.

Moving beyond visualization of known markers, we used totalVI in a more unbiased approach to quantify the differences between the B cell clusters with a one-vs-all DE test (Methods 4.12). With a single DE test, totalVI was able to provide information about these B cell subsets using both their protein and RNA expression (Figure 5f, g). As expected, the totalVI DE test found the set of six known surface markers in Figure 5b to be among the top three protein markers defined by the corresponding one-vs-all test (Figure 5f). Most RNA molecules encoding the marker proteins were also differentially expressed in the respective one-vs-all test. The complete set of differentially expressed genes, however, contained many additional molecules at similar or higher level of significance. This list included informative genes whose products are not necessarily present on the surface of the cell, such as the transcription factor Bhlhe41 that marks B1 B cells (Figure 5g, [60]).

More globally, the protein data combined with the transcriptome-wide view enabled a more refined characterization of variation within the four major sub-populations identified above by surface markers. One example of this is a sub-population of mature B cells, labeled here as Ifit3-high B cells. The Ifit3-high B cells expressed all of the protein and RNA markers of mature B cells and could not be clearly distinguished from the remaining mature B cells based on protein data alone (maximal LFC across all proteins was less than 0.19). Nevertheless, based on transcriptome-wide DE analysis, this cluster could still be distinguished as a sub-type of mature cells by the elevated expression of a host of interferon response genes (Figure 5g). This observation was also supported by a gene signature analysis with Vision [61], which identified two interferon response signatures that were enriched in the Ifit3-high B cell cluster (Methods 4.14, Supplementary Figure S15a, b). The expression of interferon response genes was not necessarily expected in this steady state condition (i.e., with no induced inflammation), however, we found the Ifit3-high B cell cluster as well as Ifit3-high T cell clusters to be represented in both biological replicates, and therefore took it to capture part of the biology in the SLN-all dataset (Supplementary Figure S15c, d).

As a second example, we explored the variability within the subset of transitional B cells and its relationship with the process of B cell development. Interestingly, latent dimension 16 (Z_{16}) captured a gradual transition within this cluster: from a small population of Rag1 expressing cells (indicating early development [58]) to cells that were closer to the mature cluster (Figure 5i, Supplementary Figure S16b, c). We used totalVI to explore how development from transitional to mature B cells may be associated with coordinated changes in gene and protein expression. To do this, we calculated the totalVI Spearman correlations separately within transitional and mature B cells for a set of features that distinguished between the two subsets (Methods 4.14). Within the transitional B cells, hierarchical clustering clearly stratified this feature set into two anti-correlated modules, one corresponding to proteins and RNA associated with transitional B cells and the other to mature B cells (Figure 5h). These modules, however, were less apparent in a hierarchical clustering of these features in mature B cells (Figure S16a), indicating that the apparent coordination may be a characteristic of the transitional state. Focusing on the transitional B cells, we found that the features in the two modules significantly correlated with the axis of maturation captured by latent dimension 16 (Supplementary Figure S16d). For cells progressing along this axis, the expression of features in the transitional module decreased and those in the mature module increased (Figure 5j,

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Methods 4.14). These results therefore point to a program of transitional B cell maturation that consists of coordinated activation and repression of multiple genes and surface proteins, leading to a gradual transition in cell state.

Taken together, these analyses demonstrate the ability of totalVI to reveal biologically-relevant patterns in feature expression and to identify differentially expressed RNA and surface proteins that could independently inform future experiments and analyses. While the proteins measured did not cover the entire proteome, they provide a crucial link to the surface marker literature that has traditionally defined cell types and present an opportunity to identify new markers for which we have antibodies. Many of the most defining RNA features in B cells do not encode surface proteins, but nonetheless are likely to play defining roles in cell state through transcriptional regulation. Through this multi-level analysis based on a combined RNA-protein view in a joint latent space, we could thus obtain a more complete picture of cell identity.

3 Discussion

We have developed totalVI, a scalable, probabilistic framework for end-to-end analysis of paired transcriptome and protein measurements in single cells. totalVI couples a low-dimensional representation of the paired data with disparate downstream analysis tasks such as visualization, identification of cell types and cell states, integration of datasets, data denoising, imputation of missing protein measurements, and differential expression testing. totalVI builds upon previous autoencoder-based methods that have been successfully applied to a variety of common tasks in single-cell transcriptomics data analysis [13, 24]. To handle the addition of proteins, totalVI assumes that RNA and protein measurements are generated from the same latent space of cells that captures their state – an assumption shared by a number of multi-omics methods [36, 62, 63].

A distinction of totalVI is that it explicitly models modality-specific technical factors that contribute to the observed data, obviating the need for a preprocessing step of normalization, which may induce biases [64]. For proteins specifically, totalVI leverages the concept of mixture density networks [65], establishing for each protein within each cell a negative binomial mixture in which the component with smaller mean corresponds to background. We demonstrated that this mixture could be effectively used to identify protein measurements that are the result of protein background and likewise those that correspond to real surface proteins. This enabled a denoised view of the data, which led to more accurate differential expression results.

We demonstrated how totalVI can be used to conduct a tiered analysis in our characterization of the heterogeneity of B cells in the murine spleen and lymph nodes. Based on a joint latent space containing both RNA and protein information from integrated datasets, we identified clusters, visualized and interpreted those clusters using known B cell surface markers, and performed a higher-resolution differential expression analysis that revealed a combination of RNA and protein markers that characterized each cluster. This analysis demonstrates an approach that can be applied to other biological systems to identify new cell types and surface markers, and provide a more detailed understanding of cell states and molecular processes through the combination of transcriptomic and proteomic information.

Going beyond the characterization of cell types, totalVI can uncover relationships between RNA and protein molecules within a cell. For example, totalVI could be used to investigate the relationship between the level of an RNA transcript and the level of its encoded protein in different biological settings, which remains an open question [66]. Previous works aiming to quantify RNA-protein relationships with paired measurements have reported weak positive correlations [8, 9]. However, it is unclear to what extent these low correlations can be attributed to technical noise in RNA and protein measurements or to biological factors such as transcription, translation, and post-translation dynamics. We found that the totalVI correlations were higher in magnitude than raw correlations across the majority of RNA-protein pairs, suggesting that the low correlations between RNA and proteins observed in previous studies could have been the result of technical noise. Future work quantifying correlations and regulatory relationships between RNA and protein features could inform our understanding of co-regulated gene networks, signal transduction pathways, or transcription and translation dynamics [12].

totalVI is the culmination of an iterative design process that consists of defining a candidate generative process and inference procedure, and then criticizing the model's performance [22]. This process

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

encodes domain knowledge into the hierarchical model that disentangles sources of variation. In modeling the protein data, we added components like protein background parameters that reflect our understanding of the CITE-seq experimental data-generating process, which we further discuss in Appendix A. Furthermore, we addressed a number of practical concerns with neural-network-based approaches, including their interpretability and sensitivity to hyperparameter choices (e.g., number of layers, learning rate, etc.) [67]. To address interpretability, we constrained the totalVI latent space to be the probability simplex, which enabled us to link variation in latent dimensions to particular genes and proteins with archetypal analysis [26]. This revealed that some of totalVI's latent dimensions were associated with specific cell types, while others corresponded to more global sources of variation, which could be observed within several cell types. With regard to the second concern, we used a default set of hyperparameters across all datasets and analysis tasks (Appendix C), demonstrating state-of-the-art results.

While the totalVI model was designed in consideration of the CITE-seq experimental protocol, we envision totalVI being used to inform experimental design by optimizing methods that affect signal-to-noise (Appendix A). For instance, totalVI could help identify optimal antibody titrations that improve the identification of foreground and background in protein measurements. totalVI could also be used to identify sequencing depths for RNA and protein libraries that balance the information gained per measurement in various analysis tasks with the cost of additional sequencing [68, 69].

totalVI is available as part of the `scvi` software package, thus allowing for a consistent programmatic interface and workflow for users analyzing different types of single-cell data. This software package includes tutorials that demonstrate functionality described here, as well as the interaction between totalVI and Scanpy [70], a popular Python-based single-cell analysis pipeline. Since totalVI processes datasets in mini-batches of hundreds of cells, it has a small memory footprint and therefore can typically be used with free cloud computing environments like Google Colab. Overall, the accessibility and scalability of totalVI make it readily available to contribute to and extract knowledge from growing community efforts such as the Human Cell Atlas [4].

The flexibility and scalability of totalVI make it easily applicable to future datasets with larger protein panels, and enable extensions that incorporate additional paired measurements. For example, we expect totalVI to naturally handle intracellular proteins measured with barcoded antibodies. Further additions of modalities like chromatin accessibility [71] or clonotype features [72] is made straightforward within the totalVI codebase with consideration of the modality-specific likelihood. By combining multiple views of cellular processes, totalVI could reveal a more complete picture that redefines cell states and elucidates mechanistic relationships between molecular components of the cell.

Acknowledgements

We thank Ellen Robey, Lydia Lutes, and Derek Bangs for help designing experiments. We thank the BioLegend Inc. and their proteogenomics team, especially Bertrand Yeung, Andre Fernandes, Qing Gao, Hong Zhang, Tse Shun Huang, for providing reagents and expertise and for help with sample preparation, library generation, and sequencing of CITE-seq libraries. We thank David DeTomaso for general data analysis advice, and Pierre Boyeau and Achille Nazaret for help with integrating totalVI in the `scvi` package. We thank members of the Streets and Yosef laboratories for helpful feedback. Research reported in this manuscript was supported by the NIGMS of the National Institutes of Health under award number R35GM124916 and by the Chan-Zuckerberg Foundation Network under grant number 2019-02452. A.G. is supported by NIH Training Grant 5T32HG000047-19. Z.S. is supported by the National Science Foundation Graduate Research Fellowship. A.S. and N.Y. are Chan Zuckerberg Biohub investigators.

Author contributions

A.G. and Z.S. contributed equally. A.G., Z.S., A.S., and N.Y. designed the study. A.G., Z.S, R.L., J.R., and N.Y. conceived of the statistical model. A.G. implemented the totalVI software with input from R.L. K.L.N. designed and produced antibody panels and provided input on the study. Z.S. designed and led experiments with input from A.S. and N.Y. A.G. and Z.S. designed and implemented

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

analysis methods and applied the software to analyze the data with input from A.S. and N.Y. A.S. and N.Y. supervised the work. A.G., Z.S., R.L., J.R., A.S., and N.Y. participated in writing the manuscript.

Competing interests

K.L.N. is an employee of BioLegend Inc.

4 Methods

4.1 The totalVI model

totalVI estimates a conditional distribution for cell n, $p_\nu(x_n, y_n \mid s_n)$, in which x_n is the G-dimensional vector of observed RNA counts (G genes), y_n is the T-dimensional vector of observed protein counts (T proteins) and s_n is the B-dimensional one-hot vector describing the batch index (experiment identifier). In total, there are N cells. We use ν to refer to the set of all generative parameters, which are described throughout this section. This distribution is estimated using the framework of variational autoencoders (VAE; [25]).

We begin by describing the generative process, for which a graphical summary is in Figure S17 and an algorithmic summary is in Algorithm 1. We then describe the inference procedure, as well as how downstream analysis tasks are directly linked to posterior queries of the model.

Priors The latent cell representation $z_n \sim \text{LogisticNormal}(0, I)$, where the logistic normal distribution is a distribution over the probability simplex. This specification enables cells to be interpreted with archetypal analysis (Methods 4.13). Typically in VAEs, z_n follows an isotropic normal distribution, which is chosen for computational convenience [25]. In this setting, a logistic normal distribution arises as transforming a sample from a normal distribution with a softmax function. A full description is in Appendix C. For all experiments, we set z_n to 20 dimensions. The latent RNA size factor $\ell_n \mid s_n \sim \text{LogNormal}(\ell_\mu^\top s_n, \ell_{\sigma^2}^\top s_n)$, where $\ell_\mu \in \mathbb{R}^B$ and $\ell_\sigma^2 \in \mathbb{R}_+^B$ are set to the empirical mean and variance of the log RNA library size (defined as total RNA counts of a cell) per batch. We use a protein-specific prior for the protein background intensity, where $\beta_{nt} \mid s_n \sim \text{LogNormal}(c_t^\top s_n, d_t^\top s_n)$. The parameters for the background intensity, $c_t \in \mathbb{R}^B$ and $d_t \in \mathbb{R}_+^B$, are protein specific and are treated as model parameters learned during inference. This prior is motivated by the observation that some component of the background is due to ambient antibodies. A prior can also be thought of as regularizing the posterior distribution, thus reducing the influence of outliers [73].

RNA likelihood Given z_n, ℓ_n, and s_n, an observed expression level x_{ng} follows a negative binomial distribution, which we present here as a Gamma-Poisson mixture:

$$\rho_n = f_\rho(z_n, s_n) \tag{1}$$
$$w_{ng} \mid z_n, \ell_n, s_n \sim \text{Gamma}(\theta_g, \ell_n \rho_{ng}) \tag{2}$$
$$x_{ng} \mid w_{ng} \sim \text{Poisson}(w_{ng}) \tag{3}$$

The gamma distribution is parameterized by its shape and mean. The mean is equal to $\ell_n \rho_{ng}$, where ℓ_n, a scaling factor, is multiplied by ρ_{ng}, interpreted as a normalized gene frequency (because ρ_n is nonnegative and sums to one). ρ_n is the output of a neural network f_ρ, which takes z_n and s_n as input (Algorithm 1).

Integrating out w_{ng} results in the following conditional distribution:

$$x_{ng} \mid z_n, \ell_n, s_n \sim \text{NegativeBinomial}(\ell_n \rho_{ng}, \theta_g). \tag{4}$$

The parameter θ_g, which is the shape of the gamma distribution, is also the inverse dispersion of the negative binomial. Further details on this connection are in Appendix B. We perform inference on the model with w_{ng} integrated out. We also treat θ_g as a model parameter learned during inference. Overall, this likelihood is equivalent to that presented in scVI [13], without zero-inflation.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Algorithm 1: The totalVI generative model. The gamma distribution is parameterized by its shape and mean. Let ν be the set of model parameters described here. A dataset has G genes and T measured proteins.

```
Define: Neural networks
$f_\rho(z_n, s_n) : \Delta^{K-1} \times \{0,1\}^B \to \Delta^{G-1}$,          (Softmax output activation)
$g_\alpha(z_n, s_n) : \Delta^{K-1} \times \{0,1\}^B \to [1, \infty)^T$,       (ReLU plus one output activation)
$h_\pi(z_n, s_n) : \Delta^{K-1} \times \{0,1\}^B \to (0,1)^T$                 (Sigmoid output activation)
Require: Inverse dispersion parameters $\theta \in \mathbb{R}_+^G, \phi \in \mathbb{R}_+^T$. Neural network parameters.
for each cell $n$ do
    $z_n \sim \text{LogisticNormal}(0, I)$                    $K$-dimensional cellular state variable
    $\rho_n = f_\rho(z_n, s_n)$                                $G$-dimensional RNA frequency
    $\alpha_n = g_\alpha(z_n, s_n)$                            $T$-dimensional foreground increment protein scaling
    $\pi_n = h_\pi(z_n, s_n)$                                  $T$-dimensional mixture parameter
    $\ell_n \sim \text{Lognormal}(\ell_\mu^\top s_n, \ell_{\sigma^2}^\top s_n)$      Cell scaling factor for RNA
    for each gene $g$ do
        $w_{ng} \sim \text{Gamma}\left(\theta_g, \ell_n \rho_{ng}\right)$
        $x_{ng} \sim \text{Poisson}\left(w_{ng}\right)$
    for each protein $t$ do
        $\beta_{nt} \sim \text{Lognormal}(c_t^\top s_n, d_t^\top s_n)$     Scalar background mean
        $v_{nt} \sim \text{Bernoulli}(\pi_{nt})$                   Scalar foreground/background mixing variable
        if $v_{nt} = 1$ then
            $r_{nt} \sim \text{Gamma}\left(\phi_t, \beta_{nt}\right)$
            $y_{nt} \sim \text{Poisson}\left(r_{nt}\right)$
        else
            $r_{nt} \sim \text{Gamma}\left(\phi_t, \beta_{nt}\alpha_{nt}\right)$
            $y_{nt} \sim \text{Poisson}\left(r_{nt}\right)$
```

Protein likelihood To capture observed protein counts arising from the background or foreground, we model y_{nt} with a negative binomial mixture, given z_n, β_n and s_n. This conditional distribution is described by the following process:

$$\pi_n = h_\pi(z_n, s_n) \tag{5}$$

$$\alpha_n = g_\alpha(z_n, s_n) \tag{6}$$

$$v_{nt} \mid z_n, s_n \sim \text{Bernoulli}(\pi_{nt}) \tag{7}$$

$$r_{nt} \mid v_{nt}, \beta_{nt}, z_n, s_n \sim \text{Gamma}(\phi_t, v_{nt}\beta_{nt} + (1 - v_{nt})\beta_{nt}\alpha_{nt}) \tag{8}$$

$$y_{nt} \mid r_{nt} \sim \text{Poisson}(r_{nt}) \tag{9}$$

Here v_{nt} controls which mixture component generates the counts. Its parameter, π_{nt}, is the output of the neural network $h_\pi(z_n, s_n)$. Notably, α_{nt}, which is the output of the neural network $g_\alpha(z_n, s_n)$, is greater than one. This ensures that one of the mixture components is always larger than the other, allowing us to interpret one component as background and one component as foreground. Furthermore, π_{nt} is interpreted as the probability that any cell-protein pair has observed counts due to background alone. For one mixture component, $y_{nt} \mid z_n, \beta_{nt}, s_n, v_{nt}$ follows a negative binomial distribution, as can be seen by integrating out r_{nt}. Finally, integrating out v_{nt} too shows that y_{nt} given z_n and s_n follows a negative binomial mixture distribution, where ϕ_t is a protein-specific inverse dispersion parameter.

4.2 Inference for totalVI

Here we describe the inference procedure for totalVI except for the case of missing data (i.e., totalVI union). The model evidence, $p_\nu(x_{1:N}, y_{1:N} \mid s_{1:N})$, cannot be computed as the integrals are analytically intractable, so Bayes rule cannot be directly applied to find a posterior distribution. Therefore, we use variational inference [74] to approximate the posterior distribution with a distribution having the following factorization:

$$q_\eta(\beta_n, z_n, \ell_n \mid x_n, y_n, s_n) := q_\eta(\beta_n \mid z_n, s_n) q_\eta(z_n \mid x_n, y_n, s_n) q_\eta(\ell_n \mid x_n, y_n, s_n). \tag{10}$$

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Here η is the set of parameters of an inference network, commonly called the *encoder* – a neural network that takes a cell's combined expression as input and outputs the parameters of the approximate posterior (e.g., mean and variance). Factors of the posterior approximation share the same family as their respective priors (e.g., $q(\beta_n \mid z_n, s_n)$ is lognormal). As described previously, we integrate out the latent variables v_{nt}, r_{nt} and w_{ng} (Algorithm 1), yielding $p_\nu(y_{nt} \mid z_n, \beta_{nt}, s_n)$, which is a mixture of negative binomials and $p_\nu(x_{ng} \mid z_n, s_n, \ell_n)$, which is a negative binomial distribution.

We optimize the evidence lower bound (ELBO) [74] of $\log p_\nu(x_{1:N}, y_{1:N}|s_{1:N})$ with respect to the variational parameters η and model parameters ν using stochastic gradients [25]. In other words, we learn the model parameters and posterior distributions simultaneously. In the VAE framework, the generative neural network is referred to as the *decoder*. Each iteration of training consists of randomly choosing a mini-batch of data (256 cells), estimating the ELBO based on this mini-batch, and updating the parameters via automatic differentiation operators. The terms corresponding to Kullback-Leibler divergences of the ELBO (see Appendix C) follow a deterministic warm-up scheme [75], which helps to avoid shallow local maxima. We use the Adam optimizer [76] with weight decay to update the model parameters. Learning rate reductions and early stopping are performed based on the ELBO of a validation set.

All neural networks are feedforward and and use standard activations (e.g., exponential, softmax, sigmoid) to encode the variational and generative distributions. We use the same hyperparameters for all our experiments. Appendix C gives further implementation details.

Inference in the case of missing proteins Here we adapted the training procedure from [77] to handle missing protein data. As any single batch may correspond to an experiment that used a different protein panel (or no proteins in the case of a scRNA-seq experiment), the missingness of protein features depends on the batch index s_n. Further, suppose all batches share the same set of genes. Across all batches, there are T proteins. For cell n, we denote the observed protein expressions y_n^{obs} and the unobserved protein expressions y_n^{mis}. The log likelihood of the observed data decomposes as

$$\log p_\nu(x_{1:N}, y_{1:N}^{\text{obs}}, s_{1:N}) = \sum_{n=1}^{N} \log p_\nu(x_n, y_n^{\text{obs}} \mid s_n) \tag{11}$$

The generative process for the observed data is the same as in Algorithm 1, with appropriate modification to only generate the features present in a particular batch. Thus, ν is the same set of model parameters described previously. Again, we use variational inference to approximate the posterior distribution with the distribution in Equation 10. In fact, all approximate posteriors share the same encoder parameters η. We optimize the ELBO of Equation 11 similarly to the procedure used when there is no missing data (i.e., we optimize the ELBO with respect to the model parameters ν and variational parameters η). To handle mismatched dimensions in the encoder, we substitute zeros for missing proteins, and for the decoder, we only calculate the ELBO terms corresponding to observed data [40]. Therefore, this procedure naturally extends to the case when there is no observed protein data for a cell n, which would be the case when the cell is obtained from a scRNA-seq experiment. Since the quality of missing protein imputation depends on (i) the goodness of fit of totalVI to the protein for the data in which it was observed and (ii) the statistical distance of the aggregated posterior distributions of z_n for each of the batches [77, 78], we add a domain adaptation regularization term to the ELBO when training [79]. A scaling factor on this regularization term decays from one to zero early in training.

4.3 Posterior predictive distributions linked to downstream tasks

For tasks like differential expression, denoising, and finding correlations, totalVI estimates functionals of posterior predictive distributions [22]. Define $C_n = \{x_n, y_n, s_n\}$ as the set of observed data for cell n. First, consider the connection between the posterior predictive distribution of RNA data to totalVI denoised RNA expression. The posterior predictive RNA expression x_{ng}^* for gene g given C_n is distributed following:

$$p(x_{ng}^* \mid C_n) \approx \int p_\nu(x_{ng}^* \mid z_n, \ell_n, s_n) q_\eta(z_n, \ell_n \mid C_n) dz_n d\ell_n, \tag{12}$$

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

To produce denoised RNA expression, we compute the posterior predictive mean of x^*_{ng}. To further control for variation due to ℓ_n, we condition on $\ell_n = 1$. By the law of total expectation,

$$\mathbb{E}_{p(x^*_{ng} \mid C_n, \ell_n=1)}[x^*_{ng}] = \mathbb{E}_{q_\eta(z_n \mid C_n)}\left[\mathbb{E}_{p_\nu(x^*_{ng} \mid z_n, s_n, \ell_n=1)}[x^*_{ng}]\right] \tag{13}$$

$$= \mathbb{E}_{q_\eta(z_n \mid C_n)}[\rho_{ng}], \tag{14}$$

where ρ_{ng} is the expectation of the RNA likelihood with the additional condition that $\ell_n = 1$.

For each cell n, we can compute the denoised RNA expression by averaging samples of ρ_n generated by the following process:

1. Sample z_n from $q_\eta(z_n \mid C_n)$
2. Set $\rho_n = f_\rho(z_n, s_n)$

There are two important considerations for these posterior predictive distributions. First, we use the approximate posterior as a surrogate for the posterior. Second, these posterior predictive distributions are not tractable to compute in closed form, so we can only sample from them with ancestral sampling. Functionals of the posterior are computed using Monte Carlo integration.

Denoised protein expression After training the model, we can generate "denoised" protein expression – protein expression effectively absent of background and controlled for sampling noise. Consider the perturbed protein generative process in which we set the background intensity to zero:

$$v_{nt} \mid z_n, s_n \sim \text{Bernoulli}(\pi_{nt}) \tag{15}$$

$$\tilde{r}_{nt} \mid v_{nt}, \beta_{nt}, z_n, s_n \sim \begin{cases} \text{Gamma}(\phi_t, \beta_{nt}\alpha_{nt}) & \text{if } v_{nt} = 0 \\ \delta_0 & \text{if } v_{nt} = 1 \end{cases}. \tag{16}$$

Here δ_0 is a point mass distribution at 0. After marginalizing out v_{nt}, $\tilde{r}_{nt} \mid z_n, s_n, \beta_{nt}$ follows a zero-inflated Gamma distribution with mean $(1 - \pi_{nt})\beta_{nt}\alpha_{nt}$.

For denoising, we return the posterior predictive mean of $\tilde{r}_{nt}$. Indeed, the posterior predictive mean is equal to $(1 - \pi_{nt})\beta_{nt}\alpha_{nt}$ averaged over many posterior samples of $q(\beta_{nt}, z_n \mid C_n)$. In other words, we return the foreground mean, weighted by the probability that the observation was derived from the foreground. This can also be stated as subtracting the expected background from the expected total signal.

Missing protein imputation To impute protein expression y^*_{nt} for cell n and protein t missing in batch s_n, but that is observed in a batch $s' \neq s_n$, do the following:

1. Sample z_n from $q_\eta(z_n \mid C_n)$
2. Sample β_{nt} from $q_\eta(\beta_{nt} \mid z_n, s = s')$
3. Sample y^*_{nt} from $p_\nu(y^*_{nt} \mid z_n, \beta_n, s = s')$

This process returns samples of $p(y^*_{nt} \mid C_n, s = s')$. Intuitively, we encode the cell into the latent space, which is designed to mix the batches (i.e., be an integrated low-dimensional representation of the data), and obtain the parameters for the protein likelihood (decode) conditioned on the cell being in batch $s = s'$. Thus, the quality of imputation relies on how well batches mix in the totalVI latent space. Ultimately, we report the expected value of the imputed distribution

$$\mathbb{E}_{p(y^*_{nt} \mid C_n, s=1)}[y^*_{nt}] = \mathbb{E}_{q_\eta(z_n \mid C_n)}\left[\mathbb{E}_{p(y^*_{nt} \mid z_n, s=1, \beta_{nt})}[y^*_{nt}]\right] \tag{17}$$

We may also impute the denoised expression, by exchanging $p_\nu(y^*_{nt} \mid z_n, \beta_n, s)$ with $p_\nu(\tilde{r}_{nt} \mid z_n, \beta_n, s)$. This change would additionally remove the protein background contribution to the prediction.

Differential expression With a single model fit, totalVI can detect differentially expressed features between sets of cells, i.e., the model does not need to be retrained for every test. Here we use the Bayesian framework of [47] to detect differential expression (DE) of genes and proteins. Let

$$\lambda_{a,b} := \Lambda(z_a, z_b, s_a, s_b) := \log_2 \rho_a - \log_2 \rho_b \tag{18}$$

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

be the log fold change (LFC) of RNA expression between cells a and b. Then the probability that gene g is differentially expressed (DE) is

$$p(|\lambda^g_{a,b}| \geq \delta \mid C_a, C_b) \approx \int \mathbb{1}\{|\lambda^g_{a,b}| \geq \delta\} q(z_a \mid C_a) q(z_b \mid C_b) dz_a dz_b, \tag{19}$$

where δ is a threshold for the effect size. Intuitively, we are measuring the fraction of posterior samples that the absolute LFC greater than or equal to δ. For all experiments we set $\delta = 0.2$. We compare the DE probability to the probability that the LFC is in the null region $|\lambda^g_{a,b}| < \delta$ using a Bayes factor:

$$\mathrm{BF}^g_{a,b} = \frac{p(|\lambda^g_{a,b}| \geq \delta \mid C_a, C_b)}{p(|\lambda^g_{a,b}| < \delta \mid C_a, C_b)}. \tag{20}$$

This can also be extended to groups of cells. Let $A = \{a_1, a_2, ..., a_m\}$ be the indices of one subpopulation of interest, and $B = \{b_1, b_2, ..., b_n\}$ be the other subpopulation of interest. We then exchange the posterior distributions in Equation 19 with the aggregated posterior:

$$q_\eta(z_a \mid C_A) q_\eta(z_b \mid C_B) = \left[\frac{1}{|A|} \sum_{a \in A} q_\eta(z_a \mid C_a)\right] \left[\frac{1}{|B|} \sum_{b \in B} q_\eta(z_b \mid C_b)\right]. \tag{21}$$

In this sampling procedure, a cell representation z_a (resp. z_b) is sampled given one randomly chosen cell in subpopulation A (resp. subpopulation B). Then, it is determined if $|\lambda^g_{a,b}| \geq \delta$ via an indicator function. The DE probability is estimated based on many samples.

Furthermore, by integrating over the batch variable s_n, we effectively compare cells as if they were in the same batch [13]. For genes, this is equivalent to computing

$$p(|\lambda^g_{a,b}| \geq \delta \mid C_a, C_b) \approx \sum_{s'} \int \mathbb{1}\{|[\Lambda(z_a, z_b, s', s')]^g| \geq \delta\} p(s') q(z_a \mid C_a) q(z_b \mid C_b) dz_a dz_b. \tag{22}$$

Here $p(s')$ is a uniform prior over batches. Every time we sample from the posterior, we decode the samples using the same batch indicator, averaging the DE probability over every possible batch indicator.

For proteins, we use the same framework, but define

$$\gamma^t_{a,b} = \log_2 \left(\mathbb{E}[\tilde{r}_{at} \mid \beta_{at}, v_{at}, z_a] + \epsilon\right) - \log_2 \left(\mathbb{E}[\tilde{r}_{bt} \mid \beta_{bt}, v_{bt}, z_b] + \epsilon\right), \tag{23}$$

where the conditional expectation is equal to

$$\mathbb{E}[\tilde{r}_{at} | \beta_{at}, v_{at}, z_a] = \beta_{at} \alpha_{at} (1 - v_{at}). \tag{24}$$

This is interpreted as the foreground mean if the cell was generated from the foreground, and zero otherwise. The added constant ϵ is a "prior count" that helps define the log fold change when $\mathbb{E}[\tilde{r}_{nt} \mid \beta_{nt}, v_{nt}, z_n] = 0$. For all analysis, we set $\epsilon = 0.5$. As with genes, we are interested in calculating $p((|\gamma^t_{a,b}| \geq \delta \mid C_a, C_b)$, where in this case we integrate with respect to the distribution

$$\prod_{i \in \{a,b\}} p(v_{it} \mid z_i) q(\beta_{it} \mid z_i, s_i) q(z_i \mid C_i). \tag{25}$$

We consider features with a $\log(\mathrm{BF}) > 0.7$ as differentially expressed. This is roughly equivalent to calling features significant if the odds ratio (here equivalent to a Bayes factor) is greater than 2. Finally, we use the posterior samples of $\lambda_{a,b}$ (resp. $\gamma_{a,b}$ for proteins) as the estimate of effect size for each gene (resp. protein). Specifically, we use the median of the samples, which is robust to outliers and is also the Bayes estimator under L_1 loss.

Denoised correlation matrix construction We seek a feature-feature correlation matrix (e.g., gene-gene correlations, gene-protein cross-correlations) that summarizes biological variation, instead of technical variation. As totalVI explicitly models nuisance factors (for genes as well as proteins), we can query the model while controlling for this nuisance variation. Furthermore, because naive computations of correlations on denoised values (parameters of conditional distributions) were shown

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

to induce spurious gene-gene correlations [37], we develop a novel sampling scheme that helps remove technical variation while avoiding such artifacts.

In order to ensure our correlation matrix does not include variation from the modeled technical factors, we perturb the data generating process to fix the library size ($\ell_n = 10000$) as well as incorporate the denoised protein expression conditional distribution. In particular, we compute a correlation matrix using samples from the distribution

$$p(\log w_n, \log \tilde{r}_n \mid C_{1:N}, \ell_{1:N}). \tag{26}$$

This is also a posterior predictive density whose samples are generated with ancestral sampling. As $\tilde{r}_n$ is zero-inflated, we add the same "prior count" before taking the logarithm. For this distribution, we sample ancestrally using the aggregated posterior

$$q_\eta(z_n, \beta_n \mid C_{1:N}) = \frac{1}{N}\sum_{n=1}^{N} q_\eta(z_n \mid C_n) q_\eta(\beta_n \mid z_n, s_n), \tag{27}$$

One could in principle replace the aggregated posterior with the prior in case of analyzing dataset-wide correlations. However, this approach is more flexible as it can be applied to calculate the correlation matrix for a subpopulation $A = \{a_1, a_2, ..., a_m\}$, where A is the set of indices for the subpopulation, by conditioning the distribution on x_A and y_A.

The distinction between this procedure and those that induced spurious correlations is that the latter effectively estimates a correlation matrix using the expected value of the posterior predictive distribution, rather than estimating the correlation matrix of the posterior predictive distribution.

Out-of-batch generalization totalVI learns a transformation from z_n and s_n to the parameters of the conditional distributions for each feature (decoder). In an out-of-batch prediction, we predict the expression of a cell (e.g., the mean of conditional distribution) given a batch $s \neq s_n$. Here we describe a general way to sample posterior quantities for a cell while also "transforming" it into a different batch that was also observed for other cells. Special cases of this have already been described in the protein imputation and differential expression sections. Consider, for instance, the RNA counts in cell n and gene g. We can calculate posterior predictive samples of x_{ng} while conditioning on any arbitrary observed batch b. Then,

$$p(x^*_{ng} \mid C_n, s = b) \approx \int p_\nu(x^*_{ng} \mid z_n, s = b) q_\eta(z_n \mid C_n) dz_n. \tag{28}$$

Furthermore, we can integrate over the choice of batch by sampling from

$$\sum_b p(x^*_{ng} \mid C_n, s = b) p(s = b), \tag{29}$$

where $p(s)$ is a uniform prior over batches. We take the expected value of this particular distribution as batch-corrected, denoised gene expression data. This "transforming" can also be applied to other likelihood parameters like π_n.

4.4 CITE-seq experiment on mouse spleen and lymph node

Table 1 shows a summary of the experimental design that generated the mouse spleen and lymph node CITE-seq datasets. Below, we describe in further detail how these datasets were collected and processed.

Cell preparation Two female C57BL/6 (B6) mice at 5 weeks of age were euthanized using CO2. From each mouse, six lymph nodes were harvested, pooled in RPMI + 10% FBS media on ice, mechanically dissociated with a syringe plunger, and passed through a 70 μm strainer to generate a single cell suspension. Likewise, the spleen was harvested, placed in RPMI + 10% FBS media on ice, mechanically dissociated with a syringe plunger, and passed through a 70 μm strainer to generate a single cell suspension. For the spleen, red blood cells were lysed in Red Blood Cell Lysis Buffer (BioLegend #420302) following the manufacturer's protocol. All animal care and procedures were carried out in accordance with guidelines approved by the Institutional Animal Care and Use Committee at BioLegend, Inc.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Dataset name	Antibody panel	Day	Mouse	Tissue	Cells (captured)	Cells (post-filtering)
SLN111-D1	111	1	A	Spleen; Lymph Node	11,160	9,264
SLN111-D2	111	2	B	Spleen; Lymph Node	9,017	7,564
SLN208-D1	208	1	A	Spleen; Lymph Node	10,777	8,715
SLN208-D2	208	2	B	Spleen; Lymph Node	8,921	7,105

Table 1: Summary of spleen and lymph node datasets. Each dataset was processed in a separate 10x lane. Each day indicates a 10x run. Cells captured is the number of cells reported by Cell Ranger.

Antibody panel preparation We prepared panels containing 111 and 208 antibodies from BioLegend in the TotalSeq-A format (Supplementary Data: Antibodies). We performed a buffer exchange on each panel using a 50kDa Amicon spin column (Millipore #UFC505096) following the manufacturer's protocol to transfer antibodies into RPMI + 10% FBS. Spleen and lymph node cell suspensions were stained with different hashtag antibodies [28].

CITE-seq protocol and library preparation The CITE-seq experiment was performed following the TotalSeq protocol with two slight modifications. First, the 10 minute centrifugation at 14,000g to remove antibody aggregates was conducted prior to buffer exchange. Second, cells were stained, washed, and resuspended in RPMI + 10% FBS to maintain viability. After staining, washing, and counting, 12,000 spleen cells and 12,000 lymph node cells were mixed and loaded into a single 10x lane. We followed the 10x Genomics Chromium Single Cell 3' v3 protocol to prepare RNA, antibody-derived-tag (ADT) and hashtag-oligo (HTO) libraries [80].

Sequencing and data processing RNA, ADT, and HTO libraries were sequenced with an Illumina NovaSeq S1. Reads were processed with Cell Ranger v3.1.0 with feature barcoding, where RNA reads were mapped to the mouse mm10-2.1.0 reference (10x Genomics, STAR aligner [81]) and antibody reads were mapped to known barcodes (Table 2). Hashtags were demultiplexed separately for each 10x lane with `HTODemux` in Seurat v3 using the kmeans function [38]. No read depth normalization was applied when aggregating datasets.

Name	RNA reads	Protein reads	RNA UMI	Protein UMI
SLN111-D1	34,717	4,733	4,392	2,785
SLN111-D2	45,765	6,542	2,121	3,419
SLN208-D1	33,569	5,513	4,561	2,956
SLN208-D2	43,821	3,961	2,102	2,261

Table 2: Sequencing statistics for spleen and lymph node datasets. Sequencing statistics calculated per 10x lane by Cell Ranger. RNA reads: mean reads per cell from RNA. Protein reads: mean reads per cell from antibody barcodes. RNA UMI: median UMI counts per cell from RNA. Protein UMI: median UMI counts per cell from antibody barcodes.

4.5 Additional datasets

We also used publicly available CITE-seq datasets from 10x Genomics. These included "10k PBMCs from a Healthy Donor - Gene Expression and Cell Surface Protein" (PBMC10k, [31]), "5k Peripheral blood mononuclear cells (PBMCs) from a healthy donor with cell surface proteins (v3 chemistry)" (PBMC5k, [45]), and "10k Cells from a MALT Tumor - Gene Expression and Cell Surface Protein" (MALT, [32]).

4.6 CITE-seq data pre-processing

For each dataset, after initial cell and gene filtering, we retained at least the top 4,000 highly variable genes (HVGs) as defined by the Seurat v3 method, merging HVGs from different batches when appropriate [38]. Dataset specific filtering is described below.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Spleen and lymph node An initial cell filter removed cells expressing fewer than 200 genes. Cells labeled as either doublets or negative for hashtag antibodies by `HTODemux` were also removed. A protein library size filter retained cells with between 400 and 10,000 total protein UMI counts. We also filtered on the number of proteins detected. For cells stained with the 111 antibody panel, we removed cells with fewer than 90 proteins detected, while the cutoff was set to 170 for cells stained with the 208 antibody panel. Cells with a high percentage of UMIs from mitochondrial genes (15% or more of the cell's total UMI count) were removed. An initial gene filter removed genes expressed in 3 or fewer cells in any given batch. In addition to the top 4,000 HVGs selected by the Seurat v3 method, we retained genes that encode the proteins targeted by the 111 antibody panel. This resulted in 4,005 total genes. After all filters, the distribution of cells per dataset was: (SLN111-D1, 9,264 cells), (SLN111-D2, 7,564 cells), (SLN208-D1, 8,715 cells), (SLN208-D2, 7,105 cells). This is a total of 32,648 cells. Unless otherwise stated, we filtered out isotype control antibodies (9 total in the 208 panel) and hashtag antibodies. The protein CD49f was also removed due to having very low total UMI counts.

PBMC10k, PBMC5k, & MALT For each of these datasets, we first removed doublets using DoubletDetection [82]. Cells with high mitochondrial content (percentage of UMIs from mitochondrial genes), high number of genes detected, high UMI counts, and with fewer than 200 genes expressed were removed. Next, cells with outlier protein library size (on either end) were removed. Genes with expression in three or fewer cells were removed. Finally, the top 4,000 HVGs were retained. Dataset specific parameters are in Table 3. In the case where the PBMC datasets are integrated, the 4,000 HVGs are selected by merging HVGs computed on each dataset separately as in the Seurat v3 method.

Dataset	No. cells	Pct. Mito	Protein Lib Size Range	No. Genes Expr.	RNA lib size
PBMCK10k	6,855	$< 10\%$	[400, 20,000]	$< 4{,}500$	$< 20{,}000$
PBMC5k	3,994	$< 20\%$	[400, 20,000]	$< 4{,}500$	$< 20{,}000$
MALT	6,838	$< 15\%$	[400, 20,000]	$< 5{,}000$	$< 30{,}000$

Table 3: Summary of filtering parameters for publicly available datasets. Ranges indicate criteria for retained cells.

4.7 Posterior predictive checks and held-out metrics

Posterior predictive checks are useful to check the fit of Bayesian models. They work by comparing the observed data to posterior predictive samples from the model [29]. Much of the benchmarking done here was inspired by previous work done to benchmark the scHPF model [14]. We primarly compared totalVI to factor analysis, which is a linear-Gaussian alternative to totalVI, and is easily extendable to multiple modalities are features are treated conditionally independent of the latent representation. We also compared performance on RNA only to scVI [13]. Posterior predictive samples for totalVI and scVI were obtained by calling the `generate` function in the scVI package. We ran scVI with 20 latent dimensions and negative binomial conditional distribution in order to be consistent with totalVI. Factor analysis (FA) models were fit using the sklearn package [83] on the combined RNA and protein measurements using one of two normalization procedures. The first procedure consisted of transforming each value by log(count + 1). The second procedure consisted of log library size normalizing the RNA features and protein features separately. For example, considering only the RNA measurements for a cell, we normalized each cell to sum to 1 by dividing by the library size of RNA, multiplied by 10,000, added 1 to each value, and took a log transformation:

$$\tilde{x}_{ng} = \log\left(L\frac{x_{ng}}{\sum_g x_{ng}} + 1\right), \tag{30}$$

where $L = 10000$. This process was then applied to the protein measurements. We refer to this type of normalization as *log library size normalization*, and for short, *log rate*. These normalization procedures are necessary as FA assumes a Gaussian distribution, so training on the raw data would lead to poor model fit. Posterior predictive samples for FA models were computed using the fitted

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

parameters and posterior distribution derived in [84]. We note that normalization procedures were inverted so that FA posterior predictive samples were on the same scale as the raw data.

For each dataset, each model was trained on a train set comprising of 85% of the cells. An additional 5% of cells were held-out as a validation set for totalVI early stopping. The remaining 10% of cells were also held-out as a test set. For each model's posterior predictive samples (25 for each model) based on the train set, we calculated the coefficient of variation (CV) for each feature, and calculated the mean absolute error between the average CV and the observed raw data CV. We also used posterior predictive samples to assess generalization to unseen data. In this setup, we generated posterior predictive samples (150 for each model) conditioned on the test set. We considered the mean absolute error between the observed held-out data and the posterior predictive mean.

Moreover, we computed a held-out calibration error [30] for each model based on the test set. For each cell n and gene g, let I_{ng} be the indicator that the observed value is contained in the interval of all posterior predictive samples. The calibration error for genes is then calculated as

$$\mathrm{Cal}^{\mathrm{RNA}} = \left(1 - \frac{1}{NG}\sum_n \sum_g I_{ng}\right)^2. \tag{31}$$

The calibration error for proteins is computed separately following the same procedure.

Finally, for totalVI and scVI only, and for only the RNA data, we computed the held-out predictive log likelihood. In this metric, z_n and ℓ_n were sampled from the variational posterior for each cell n and the average negative conditional log likelihood, $-\log p(x_n \mid z_n, \ell_n, s_n)$ was computed. This is also called the reconstruction loss in the VAE literature. This is also an approximation of $-\log p(x_n \mid x_n, y_n, s_n)$, the negative predictive log likelihood. We note that we cannot compare the log likelihood of totalVI and scVI, which use discrete conditional distributions to factor analysis models, which use continuous conditional distributions.

4.8 Background decoupling benchmarking

We reported the totalVI background probability as the posterior predictive mean of π_{nt}, thus

$$p(\text{cell } n, \text{ protein } t \text{ is background}) = \mathbb{E}_{p(\pi_{nt} \mid x_n, y_n, s_n)}[\pi_{nt}], \tag{32}$$

where the expectation is approximated using Monte Carlo integration. The totalVI foreground probability is one minus the background probability.

Observing protein background in empty droplets and non-expressing cell types To observe different sources of protein background, we considered both empty droplets and cell types with known expression of surface markers. We defined empty droplets as non-cell barcodes from the SLN111-D1 dataset with between 20 and 100 RNA UMI counts (approximately 75,358 barcodes). We chose these criteria so that empty droplets were likely to represent ambient molecules rather than sequencing errors (with very low UMI counts) or cell debris (with higher UMI counts) [85]. To observe non-specific binding of antibodies, we considered B cells (which are known to express CD19 and CD20, but not CD28) and T cells (which are known to express CD28, but not CD19 or CD20). Using cell type annotations as described below (Methods 4.11), we grouped all high-quality, non-doublet B cell clusters (excluding plasma B cells), and alpha/beta T cell clusters (including all CD4, Treg, and CD8 T cell clusters). We observed that for these three proteins, both empty droplets and the non-expressing cell type contained protein background (non-zero protein counts) with varying degrees of overlap with the foreground signal of the expressing cell type. In this text, we describe the protein counts of the non-expressing cell type above the counts in empty droplets as non-specific antibody binding, although we acknowledge there could be multiple sources of this cell-specific background (Appendix A).

GMM cutoff for protein foreground/background As a baseline determination of a cutoff to distinguish cells with foreground or background protein expression, we used a Gaussian mixture model (GMM). We applied scikit-learn's `GaussianMixture` with default parameters to fit a GMM with two components to the log(protein counts + 1) for each protein for all cells in the SLN111-D1 dataset. We interpreted the posterior probability of the component with the higher mean as the foreground probability and that of the lower mean as the background probability. The GMM cutoff

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

between foreground and background was determined to be the protein expression level at which the foreground probability was closest to 0.5.

Classification of cell type by marker proteins We sought to evaluate totalVI against a GMM at predicting major cell types by the foreground probability of commonly used surface markers. Restricting all cells to just those that fell into the categories of B cells or T cells as described above, we tested how well totalVI or a GMM could classify cell types based on commonly used protein surface markers. We used a GMM fit on all cells of the SLN111-D1 dataset for each protein as described above. For each protein, we computed a receiver operating characteristic curve (ROC) (`sklearn.metrics`) by thresholding the totalVI or GMM foreground probability estimates (described above), using cell type annotations as true labels. We reported the area under the ROC (ROC AUC). The cell type considered as the positive population was either B cells, T cells, CD4 T cells, or CD8 T cells depending on the marker. In tests considering each of these positive populations, all remaining cells among the B and T cells were considered the negative population. Marker proteins tested included, for B cells: CD19, CD45R-B220, CD20, I-A-I-E (MHC II); for T cells: CD5, TCRb, CD28, CD90.2; for CD4 T cells: CD4; for CD8T cells: CD8a, CD8b [86–89]. Although we are aware of documented exceptions to these markers appearing strictly on a single cell type (e.g. CD5 is expressed on a portion of B1 B cells), these exceptions are rare. In these cases where marker expression is not mutually exclusive, cell types can still be distinguished by the gradation in levels of the marker between cell populations. Thus, these exceptions do not negate the utility of these markers in broad cell type classification (which is apparent in both totalVI and GMM performance at this classification task).

Visualization and raw data normalization For the SLN111-D1 dataset, we visualized the totalVI latent space in two dimensions using Scanpy's [70] implementation of the UMAP algorithm [44]. We applied log library-size normalization to the raw RNA counts as in Equation 30.

Distinguishing foreground and background in trimodal protein distributions Despite using a two-component mixture, totalVI can decouple the background and foreground of proteins that have more than two modes globally. totalVI is capable of distinguishing foreground and background in this setting because the mixture is conditionally dependent on z_n, which allows the foreground and background expression modes to be defined locally in the latent space. For example, as has been reported using flow cytometry [90], CITE-seq data of peripheral blood mononuclear cells contains three modes of CD4 expression corresponding to CD4 T cells, monocytes, and background. totalVI detected that both CD4 T cells and monocytes had foreground expression of CD4, while the CD4 expression of the remaining cells was attributable to background expression.

Denoised protein expression Denoised protein expression was calculated as described in Methods 4.3. B cells and T cells were defined by annotations, as described above.

4.9 RNA-protein correlation analysis

Evaluation of correlation calculation with permuted features Using totalVI, we aimed to calculate a correlation matrix between all RNA and protein features free from nuisance variation such as sequencing depth and protein background. We took care to avoid the naive calculation of correlations directly between denoised features, noting that a recent study reported false positive correlations in smoothed scRNA-seq data [37]. Instead, we developed a novel sampling method for the calculation of denoised feature correlations that removes nuisance variation while avoiding imputation-induced artifacts (Methods 4.3).

To evaluate whether totalVI could calculate a denoised feature correlation matrix without introducing spurious relationships in the data, we permuted the expression of a set of genes to serve as a negative control. To create this set of negative control genes from the SLN111-D1 dataset, we selected the 100 genes with highest mean expression that were not already among the top highly variable genes used in the model (Methods 4.6). We randomly permuted the counts of these genes within each cell, rendering these genes independent of all other gene and protein features. After concatenating the SLN111-D1 dataset with the permuted gene expression for all cells, we ran the totalVI model.

We then calculated Pearson and Spearman correlations between features using three methods, referred to here as raw, naive totalVI, and totalVI correlations. Raw correlations were calculated

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

between log library-size normalized RNA (Equation 30) and log(protein counts + 1). Naive totalVI correlations were calculated between totalVI denoised RNA (Methods 4.3) and totalVI denoised proteins (Methods 4.3). totalVI correlations were calculated by sampling denoised RNA and denoised protein values from the posterior (Methods 4.3).

We observed the correlations between all RNA and protein features as well as the 100 additional genes whose expression was randomly permuted. By comparing the raw correlations with denoised correlations, we observed whether the method of denoising could maintain the relationship between these permuted genes and other features, which, in expectation, are independent from each other. Here, we highlighted the correlations between all proteins and the randomly permuted genes, whose correlations are expected to be near zero.

Correlations of RNA-protein pairs We calculated a feature correlation matrix for the SLN111-D1 dataset using either the totalVI sampling method or by calculating raw correlations as described above. The resulting feature correlation matrices for both Pearson and Spearman correlations were subset to each protein and its encoding RNA for all proteins with a unique encoding RNA in the dataset (i.e., excluding RNA with multiple isoforms such as Ptprc).

4.10 Integration of multiple datasets

We compared totalVI's integration performance to that of Scanorama [39] and Seurat v3 [38]. We chose these methods because like totalVI, they provide both batch-corrected measurements and a low-dimensional integrated latent space. The input to both Scanorama (`scanorama.correct`) and Seurat v3 (`FindIntegrationAnchors`, `IntegrateData`) methods was a normalized matrix of concatenated genes and proteins. Genes were subset to the same subset used as input to totalVI. The RNA counts of this matrix were normalized following standard log library size normalization (Equation 30). For proteins, we used a $y \rightarrow \log(y+1)$ transformation. Finally, we standard scaled each feature. Scanorama and Seurat v3 were run with default parameters. We compared the performance of totalVI, Scanorama, and Seurat v3 using the following metrics:

Latent mixing metric The latent mixing metric measures how well the latent cell representations are mixed between batches relative to the global frequency of batches. First, a cell-cell similarity matrix is computed from a latent representation of cells. Next, select 100 cells uniformly at random, and calculate the frequency of batches represented in each cell's 100 nearest neighbors. Let $p_i^{(n)}$ be the frequency of batch i in the 100 nearest neighbors of cell n. Let q_i be the global frequency of batch i. Compute the negative relative entropy between the frequency of observed batches in the neighborhood, and the global frequency of batches:

$$\mathrm{KL}\left(p^{(n)} \parallel q\right) = \sum_{i=1}^{B} p_i^{(n)} \log \frac{p_i^{(n)}}{q_i} \tag{33}$$

Repeat this 50 times and return the average negative relative entropy.

Measurement mixing metric The measurement mixing metric describes how well the high-dimensional measurements are batch corrected, and for each feature, is related to the Mann-Whitney U statistic. Consider one feature in the batch-corrected data matrix placed in rank order. Let R_1 be the sum of the ranks of the cells in batch 1 and N_1 be the number of cells in batch 1. Define $U_1 = R_1 - \frac{N_1(N_1+1)}{2}$. Similarly, compute U_2 for batch 2 and return $\min(U_1, U_2)$. Higher values of this metric indicate better mixing within that feature.

Feature retention metric The feature retention metric describes how spatial autocorrelation of both RNA and protein change when comparing cells from an integrated latent representation to a latent representation derived from each batch separately. Lower values of this metric indicate that the integration procedure reduced the localization of feature expression, indicating some degree of random mixing. We calculate it as follows. For two batches and a particular integration method, we calculate Z_1 and Z_2, the latent representations of the cells of batch 1 and batch 2, respectively. The latent space computation of the individual batches was chosen to closely match the integration method (see below). We also calculate an integrated latent representation of both batches $Z^\top = [\tilde{Z}_1 \;\; \tilde{Z}_2]$. Let $D_1 = [X_1 \;\; Y_1]$ be the combined RNA and protein batch 1 in which RNA is library size log normalized

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

and proteins are log-transformed. Let $\mathbb{E}[H(D_1, Z_1)]$ be the expected feature autocorrelation score as calculated by Hotspot [43]. Furthermore, let $\mathbb{E}[H(D_1, \tilde{Z}_1)]$ be the analogous quantity calculated using the latent cell representations of batch 1 subsetted from the joint, integrated representation. The feature retention metric is calculated as $\frac{1}{2}\sum_i^2 \mathbb{E}[H(D_i, \tilde{Z}_i)] - \mathbb{E}[H(D_i, Z_i)]$. In the case of totalVI union, features were intersected to compute this metric.

For Scanorama, we define Z_1 and Z_2 to be a 100-dimensional matrix produced with principal components analysis (PCA), which is the same dimension reduction used in the integration method. For Seurat v3, we similarly use PCA to reduce D_1 and D_2 to 30 dimensions, the same number of dimensions used for integration. The input to PCA was the same as the input for the respective method, except for Scanorama, where we additionally L_2 normalized each cell, because this step is done automatically by Scanorama's `correct` method.

Missing protein imputation For Seurat v3, we imputed proteins based on mutual nearests neighbors in the RNA data using the `FindTransferAnchors` and `TransferData` functions. Again, RNA data were log library size normalized. Proteins were not normalized as input to Seurat. For totalVI, after fitting the model, cells from the batch with held-out proteins were decoded conditioned on being in the batch with observed protein data. We used the Pearson correlation of values on the log scale to assess imputation accuracy. We note that CTP-net [91] is another method for imputing protein expression from scRNA-seq data; however, at the time of writing, the CTP-net package only offers a neural network that was pre-trained on specific CITE-seq datasets from human cells, with no option training with a new dataset from either mice or humans. Therefore, a direct comparison to totalVI and Seurat v3 is not straightforward.

4.11 Stratification of cells in SLN-all

We stratified cells of the mouse spleen and lymph node based on the SLN-all dataset (totalVI-intersect model fit as described above in Section 2.6). We clustered cells in the totalVI latent space with Scanpy's implementation of the Leiden algorithm at resolution 1, resulting in 32 clusters [70, 92]. We repeated this approach to sub-cluster cells, finding a total of 43 clusters. We used Vision [61] with default parameters for data exploration, including its implementation of the Wilcoxon rank sum test, to identify cluster markers. Clusters were manually annotated based on a curated list of cell type markers (Table 4). Clusters annotated as doublets, low quality cells, or cells undergoing the cell cycle were removed from further analysis. Again, we visualized the totalVI latent space in two dimensions using Scanpy's implementation of the UMAP algorithm.

4.12 Differential expression analysis

The t-test and Wilcoxon test for each differential expression scenario were run on protein features only using the Scanpy library, which produces adjusted p-values corrected by Benjamini-Hochberg procedure [116]. A protein was considered to be differentially expressed if the adjusted p-value was less than 0.05. Each application of totalVI differential expression tests to a dataset requires a trained totalVI model. For each dataset used in DE analysis, all cells were included to train the model. Throughout, we used our manual annotations from the SLN-all totalVI-intersect model run. The cells in nuisance clusters (described in previous section) were removed before running totalVI differential expression functions.

In a given totalVI differential expression test, we identified cell type markers by first filtering features for significance (log Bayes factor > 0.7), and then sorting by the median log fold change. We only retained genes with non-zero UMI counts in at least 10% of the subset of cells.

In the comparison to scVI gene Bayes factors, each method was trained independently on the SLN111-D1 dataset. We ran scVI with 20 latent dimensions and negative binomial conditional distribution to be consistent with totalVI. Differential expression of genes in scVI was computed using the same LFC-based method, which is implemented in the `scvi` package. In reproducibility benchmarking, totalVI was trained independently on the replicates. In the test between ICOS-high Tregs and CD4 conventional T cells, we used the same totalVI-intersect model fit that was used to manually annotate the cells.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Cell type annotation	Selected markers	Selected references
Activated CD4 T cells	Itm2a, Cd69	[93]
B1 B cells	Bhlhe41, CD43, CD19	[60, 94]
CD122+ CD8 T cells	CD122, CD62L, CD183(CXCR3), CD8a	[95, 96]
CD4 T cells	CD4	[86]
CD8 T cells	CD8a, CD8b	[86]
cDC1s	Clec9a, Cd8a, Xcr1, CD11c	[97, 98]
cDC2s	Itgax, Cd4, CD11c	[97, 98]
Cycling B/T cells	Birc5, Top2a, Mki67	[99]
Erythrocytes	Hbb-bs, Hbb-bt	[100]
GD T cells	Cd3e, Tcrg-c1, Tcrg-c2, Maf, Il17re	[101]
ICOS-high Tregs	Foxp3, CD4, ICOS	[102, 103]
Ifit3-high B cells	Ifit3, CD19	
Ifit3-high CD4 T cells	Ifit3, CD4	
Ifit3-high CD8 T cells	Ifit3, CD8a	
Ly6-high monocytes	Ly6c2, Fn1, F13a1	[104]
Ly6-low monocytes	Cd36, Cd300e, Fabp4	[105]
Mature B cells	IgD, CD23, CD19	[59]
Migratory DCs	Slco5a1, Anxa3, Nudt17, Adcy6, Cacnb3	[106]
MZ B cells	CD21, CD19	[59]
MZ/Marco-high macrophages	Cd209b, Marco	[107]
Neutrophils	S100a8	[108]
NK cells	NK-1.1, Gzma, Ncr1	[86, 109]
NKT cells	Cd3e, NK-1.1, Ccl5, Klrd1	[86, 110]
pDCs	Siglech, Irf8, Runx2, CD11c	[97, 111, 112]
Plasma B cells	Jchain	[113]
Red-pulp macrophages	F4-80, C1qa, C1qb, Hmox1, Vcam1	[114]
Transitional B cells	CD93, CD24, CD19	[58, 115]
Tregs	Foxp3, CD4, CD357(GITR)	[52]

Table 4: Annotated cell types and selected markers in the spleen and lymph node datasets. cDC1: Conventional dendritic cell 1. cDC2: Conventional dendritic cell 2. GD: Gamma/delta. MZ: Marginal zone. NK: Natural killer. NKT: Natural killer T. pDC: Plasmacytoid dendritic cell. Treg: Regulatory CD4 T cell.

DE on imputed proteins In one totalVI model fit, SLN111-D1 and SLN111-D2 were integrated with the proteins of SLN111-D2 held out. In the second totalVI model fit, these two datasets were integrated with all data. In testing differential expression of proteins, and for each model fit, we conditioned on SLN111-D1. This is an application of Equation 22, except that the prior $p(s')$ is 1 if $s' =$ SLN111-D1 and 0 otherwise.

4.13 Archetypal analysis

This analysis was performed on the SLN-all intersect mode model run. As z_n is distributed as logistic normal, the latent space is then constrained to the probability simplex (i.e., each z_n is non-negative and sums to one). Archetypes correspond to vertices of the totalVI latent space, which means they can be represented by the identity matrix I_d, where d is the number of latent dimensions (20 in all experiments). In this setting, the latent space is the 19-dimensional standard simplex. We first identified and removed four archetypes from further interpretation that suffered from inactivity (a known issue in training VAEs) [117]. For the remaining 16 latent dimensions, we decoded the archetypes to obtain denoised RNA and protein archetypal expression profiles, all conditioned on batch 0 (the SLN111-D1 experiment). We then computed denoised RNA and protein expression profiles for all cells in SLN-all, conditioned on SLN111-D1. To derive signatures for each archetype, we computed the mean and standard deviation of each feature in the denoised RNA and protein expression matrices (without the archetypes) and standard scaled the archetypal profiles with respect to this mean and standard deviation. We refer to this quantity as the archetype score. The top features for each archetype were those with an archetype score greater than 2. The distance to

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

the archetype is computed as the Manhattan distance from each cell's latent representation to the archetype. The distances per archeytpe were scaled into the range $[0, 1]$.

4.14 B cell analysis

For this analysis, we used the totalVI-intersect model fit on the SLN-all dataset as described above in Section 2.6. The SLN-all dataset was filtered to include all high-quality, non-doublet clusters annotated as B cells (excluding plasma B cells), resulting in 15,560 cells (Methods 4.11).

Calculation of signature scores Gene signature analysis was conducted using Vision [61] with default parameters. Gene signatures, including interferon response signatures, were downloaded from MSigDB gene sets [118]. Signature scores were calculated on all cells in the SLN-all dataset based on cell similarities defined by the totalVI latent space.

Identification of transitional and mature B cell feature modules totalVI Spearman correlations between all features were calculated separately within the transitional B cell cluster and the mature B cell cluster. Features were subset by the following method. From a one-vs-one DE test between transitional and mature B cells, we selected the top ten marker genes and top three marker proteins for each cluster (Methods 4.12). We added to this list the four features most highly correlated with each differentially expressed feature within its respective cluster. This resulted in a list of both transitional and mature features which we used to subset the full feature correlation matrix. Features were hierarchically clustered separately for transitional and mature B cells using Seaborn's `clustermap` with default parameters.

When plotting totalVI expression of each feature as a function of $1 - Z_{16}$, each feature was standard scaled and smoothed with a loess curve. Spearman correlations were calculated between each feature and $1 - Z_{16}$. The p-values of these correlations were all significant (Benjamini-Hochberg adjusted $P < 0.001$).

Code availability

The code to reproduce the experiments of this manuscript is available at `https://github.com/YosefLab/totalVI_reproducibility`. The reference implementation of totalVI is available via the scVI package at `https://github.com/YosefLab/scVI`.

Data availability

Processed data are available in the reproducibility GitHub repository. Raw data are being uploaded to GEO. The SLN-all dataset can be explored interactively with Vision at `http://s133.cs.berkeley.edu:9000/Results.html`.

References

1. Stubbington, M. J. T., Rozenblatt-Rosen, O., Regev, A. & Teichmann, S. A. Single-cell transcriptomics to explore the immune system in health and disease. *Science* **358,** 58–63 (2017).
2. Papalexi, E. & Satija, R. Single-cell RNA sequencing to explore immune cell heterogeneity. *Nature Reviews Immunology* (2017).
3. Labib, M. & Kelley, S. O. Single-cell analysis targeting the proteome. *Nature Reviews Chemistry* **4,** 143–158 (2020).
4. Regev, A. *et al.* The Human Cell Atlas. *eLife* (2017).
5. Tanay, A. & Regev, A. Scaling single-cell genomics from phenomenology to mechanism. *Nature* (2017).
6. Todorovic, V. Single-cell RNA-seq—now with protein. *Nature Methods* **14,** 1028–1029 (2017).

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

7. Haque, A., Engel, J., Teichmann, S. A. & Lönnberg, T. A practical guide to single-cell RNA-sequencing for biomedical research and clinical applications. *Genome Medicine* **9,** 1–12 (2017).
8. Stoeckius, M. *et al.* Simultaneous epitope and transcriptome measurement in single cells. *Nature Methods* (2017).
9. Peterson, V. M. *et al.* Multiplexed quantification of proteins and transcripts in single cells. *Nature Biotechnology* (2017).
10. Wagner, A., Regev, A. & Yosef, N. Revealing the vectors of cellular identity with single-cell genomics. *Nature Biotechnology* (2016).
11. Efremova, M. & Tiechmann, S. A. Single-cell multimodal omics: the power of many. *Nature Methods* (2020).
12. Gorin, G., Svensson, V. & Pachter, L. Protein velocity and acceleration from single-cell multiomics experiments. *Genome Biology* **21,** 1–6 (2020).
13. Lopez, R., Regier, J., Cole, M. B., Jordan, M. I. & Yosef, N. Deep generative modeling for single-cell transcriptomics. *Nature Methods* **15,** 1053–1058 (2018).
14. Levitin, H. M. *et al.* De novo gene signature identification from single-cell RNA -seq with hierarchical Poisson factorization. *Molecular Systems Biology* **15** (2019).
15. Azizi, E., Prabhakaran, S., Carr, A. & Pe'er, D. Bayesian inference for single-cell clustering and imputing. *Genomics and Computational Biology* (2017).
16. Risso, D., Perraudeau, F., Gribkova, S., Dudoit, S. & Vert, J. P. A general and flexible method for signal extraction from single-cell RNA-seq data. *Nature Communications* (2018).
17. Baldwin, E. *et al.* On fusion methods for knowledge discovery from multi-omics datasets. *Computational and Structural Biotechnology Journal* (2020).
18. Bersanelli, M. *et al.* Methods for the integration of multi-omics data: Mathematical aspects. *BMC Bioinformatics* **17** (2016).
19. Granja, J. M. *et al.* Single-cell multiomic analysis identifies regulatory programs in mixed-phenotype acute leukemia. *Nature Biotechnology* **37,** 1458–1465 (2019).
20. Praktiknjo, S. D. *et al.* Tracing tumorigenesis in a solid tumor model at single-cell resolution. *Nature Communications* **11,** 991 (2020).
21. Kotliarov, Y. *et al.* Broad immune activation underlies shared set point signatures for vaccine responsiveness in healthy individuals and disease activity in patients with lupus. *Nature Medicine,* 1–12 (2020).
22. Blei, D. M. Build, compute, critique, repeat: Data analysis with latent variable models. *Annual Review of Statistics and Its Application* (2014).
23. Svensson, V. Droplet scRNA-seq is not zero-inflated. *Nature Biotechnology* (2020).
24. Eraslan, G., Simon, L. M., Mircea, M., Mueller, N. S. & Theis, F. J. Single-cell RNA-seq denoising using a deep count autoencoder. *Nature Communications* **10,** 390 (2019).
25. Kingma, D. P. & Welling, M. *Auto-Encoding variational Bayes* in *International Conference on Learning Representations* (2014).
26. Cutler, A. & Breiman, L. Archetypal analysis. *Technometrics* (1994).
27. Svensson, V., Gayoso, A., Yosef, N. & Pachter, L. Interpretable factor models of single-cell RNA-seq via variational autoencoders. *Bioinformatics* (2020).
28. Stoeckius, M. *et al.* Cell Hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. *Genome Biology* (2018).
29. Gelman, A., Meng, X. L. & Stern, H. Posterior predictive assessment of model fitness via realized discrepancies. *Statistica Sinica* (1996).
30. Kuleshov, V., Fenner, N. & Ermon, S. *Accurate uncertainties for deep learning using calibrated regression* in *International Conference on Machine Learning* (2018).
31. 10X Genomics. *10k PBMCs from a Healthy Donor - gene expression and cell surface protein* 2018. https://support.10xgenomics.com/single-cell-gene-expression/datasets/3.0.0/pbmc_10k_protein_v3.
32. 10X Genomics. *10k Cells from a MALT Tumor - gene expression and cell surface protein* 2018. https://support.10xgenomics.com/single-cell-gene-expression/datasets/3.0.0/malt_10k_protein_v3.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

33. Hulspas, R., O'Gorman, M. R., Wood, B. L., Gratama, J. W. & Sutherland, D. R. Considerations for the control of background fluorescence in clinical flow cytometry. *Cytometry Part B: Clinical Cytometry* (2009).
34. Young, M. D. & Behjati, S. SoupX removes ambient RNA contamination from droplet based single cell RNA sequencing data. *bioRxiv,* 303727 (2018).
35. Fleming, S. J., Marioni, J. C. & Babadi, M. CellBender remove-background: a deep generative model for unsupervised removal of background noise from scRNA-seq datasets. *bioRxiv.* https://doi.org/10.1101/791699 (2019).
36. Ngo Trong, T. *et al.* Semisupervised generative autoencoder for single-cell data. *Journal of Computational Biology* (2019).
37. Andrews, T. S. & Hemberg, M. False signals induced by single-cell imputation. *F1000Research* (2019).
38. Stuart, T. *et al.* Comprehensive Integration of Single-Cell Data. *Cell* **177,** 1888–1902 (2019).
39. Hie, B., Bryson, B. & Berger, B. Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. *Nature Biotechnology* (2019).
40. Mattei, P. A. & Freiisen, J. *Miwae: Deep generative modelling and imputation of incomplete data sets* in *International Conference on Machine Learning* (2019).
41. Lotfollahi, M., Naghipourfar, M., Theis, F. J. & Wolf, F. A. Conditional out-of-sample generation for unpaired data using trVAE. *arXiv.* http://arxiv.org/abs/1910.01791 (2019).
42. Haghverdi, L., Lun, A. T., Morgan, M. D. & Marioni, J. C. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nature Biotechnology* **36,** 421–427 (2018).
43. DeTomaso, D. & Yosef, N. Identifying informative gene modules across modalities of single cell genomics. *bioRxiv.* https://doi.org/10.1101/2020.02.06.937805 (2020).
44. McInnes, L., Healy, J. & Melville, J. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv.* http://arxiv.org/abs/1802.03426 (2018).
45. 10X Genomics. *5k Peripheral blood mononuclear cells (PBMCs) from a healthy donor with cell surface proteins (v3 chemistry)* 2019. https://support.10xgenomics.com/single-cell-gene-expression/datasets/3.0.2/5k_pbmc_protein_v3?.
46. Kass, R. E. & Raftery, A. E. Bayes factors. *Journal of the American Statistical Association* **90,** 773–795 (1995).
47. Boyeau, P. *et al. Deep Generative Models for Detecting Differential Expression in Single Cells* in *Machine Learning in Computational Biology* (2019).
48. Bezman, N. A. *et al.* Molecular definition of the identity and activation of natural killer cells. *Nature Immunology* **13,** 1000–1008 (2012).
49. Walzer, T. *et al.* Identification, activation, and selective in vivo ablation of mouse NK cells via NKp46. *Proceedings of the National Academy of Sciences of the United States of America* **104,** 3384–3389 (2007).
50. Gordon, S. M. *et al.* The transcription factors T-bet and Eomes control key checkpoints of natural killer cell maturation. *Immunity* **36,** 55–67 (2012).
51. Zhao, H., Liao, X. & Kang, Y. Tregs: Where we are and what comes next? *Frontiers in Immunology* (2017).
52. Roncarolo, M.-G. & Gregori, S. Is FOXP3 a bona fide marker for human regulatory T cells? *European Journal of Immunology* **38,** 925–927 (2008).
53. Fontenot, J. D., Rasmussen, J. P., Gavin, M. A. & Rudensky, A. Y. A function for interleukin 2 in Foxp3-expressing regulatory T cells. *Nature Immunology* **6,** 1142–1151 (2005).
54. Sprouse, M. L. *et al.* High self-reactivity drives T-bet and potentiates Treg function in tissue-specific autoimmunity. *JCI Insight* **3,** 1–14 (2018).
55. Townes, F. W., Hicks, S. C., Aryee, M. J. & Irizarry, R. A. Feature selection and dimension reduction for single-cell RNA-Seq based on a multinomial model. *Genome Biology* **20,** 295 (2019).
56. Korem, Y. *et al.* Geometry of the gene expression space of individual cells. *PLoS Computational Biology* **11,** 1–27 (2015).
57. Van Dijk, D. *et al.* Finding archetypal spaces for data using neural networks. *arXiv.* http://arxiv.org/abs/1901.09078 (2019).

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

58. Thomas, M. D., Srivastava, B. & Allman, D. Regulation of peripheral B cell maturation. *Cellular Immunology* **239,** 92–102 (2006).

59. Loder, F. *et al.* B cell development in the spleen takes place in discrete steps and is determined by the quality of B cell receptor-derived signals. *Journal of Experimental Medicine* **190,** 75–89 (1999).

60. Kreslavsky, T. *et al.* Essential role for the transcription factor Bhlhe41 in regulating the development, self-renewal and BCR repertoire of B-1a cells. *Nature Immunology* **18,** 442–455 (2017).

61. DeTomaso, D. *et al.* Functional interpretation of single cell similarity maps. *Nature Communications* (2019).

62. Lock, E. F., Hoadley, K. A., Marron, J. S. & Nobel, A. B. Joint and individual variation explained (JIVE) for integrated analysis of multiple data types. *Annals of Applied Statistics* **7,** 523–542 (2013).

63. Argelaguet, R. *et al.* Multi-Omics Factor Analysis—a framework for unsupervised integration of multi-omics data sets. *Molecular Systems Biology* **14,** 1–13 (2018).

64. Vallejos, C. A., Risso, D., Scialdone, A., Dudoit, S. & Marioni, J. C. Normalizing single-cell RNA sequencing data: Challenges and opportunities. *Nature Methods* **14,** 565–571 (2017).

65. Bishop, C. M. *Mixture Density Networks* tech. rep. (1994). http://publications.aston.ac.uk/id/eprint/373/.

66. Liu, Y., Beyer, A. & Aebersold, R. On the dependency of cellular protein levels on mRNA abundance. *Cell* **165,** 535–550 (2016).

67. Hu, Q. & Greene, C. S. Parameter tuning is a key part of dimensionality reduction via deep variational autoencoders for single cell RNA transcriptomics. *Pacific Symposium on Biocomputing* (2019).

68. Svensson, V., Beltrame, E. d. V. & Pachter, L. Quantifying the tradeoff between sequencing depth and cell number in single-cell RNA-seq. *bioRxiv.* https://doi.org/10.1101/762773 (2019).

69. Heimberg, G., Bhatnagar, R., El-Samad, H. & Thomson, M. Low dimensionality in gene expression data enables the accurate extraction of transcriptional programs from shallow sequencing. *Cell Systems* **2,** 239–250 (2016).

70. Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: Large-scale single-cell gene expression data analysis. *Genome Biology* (2018).

71. Clark, S. J. *et al.* ScNMT-seq enables joint profiling of chromatin accessibility DNA methylation and transcription in single cells e. *Nature Communications* **9,** 1–9 (2018).

72. Mimitou, E. P. *et al.* Multiplexed detection of proteins, transcriptomes, clonotypes and CRISPR perturbations in single cells. *Nature Methods* **16,** 409–412 (2019).

73. Wang, C. & Blei, D. M. A general method for robust Bayesian modeling. *Bayesian Analysis* (2018).

74. Blei, D. M., Kucukelbir, A. & McAuliffe, J. D. Variational Inference: A review for statisticians. *Journal of the American Statistical Association* **112,** 859–877 (2017).

75. Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K. & Winther, O. *Ladder variational autoencoders* in *Neural Information Processing Systems* (2016).

76. Kingma, D. P. & Ba, J. L. *Adam: A method for stochastic optimization* in *International Conference on Learning Representations* (2015).

77. Lopez, R. *et al. A joint model of unpaired data from scRNA-seq and spatial transcriptomics for imputing missing gene expression measurements* in *ICML workshop in Computational Biology* (2019).

78. Blitzer, J., Crammer, K., Kulesza, A., Pereira, F. & Wortman, J. *Learning bounds for domain adaptation* in *Advances in Neural Information Processing Systems* (2008).

79. Ganin, Y. *et al.* Domain-adversarial training of neural networks. *The Journal of Machine Learning Research* **17,** 2096–2030 (2016).

80. Zheng, G. X. *et al.* Massively parallel digital transcriptional profiling of single cells. *Nature Communications* (2017).

81. Dobin, A. *et al.* STAR: Ultrafast universal RNA-seq aligner. *Bioinformatics* (2013).

82. Gayoso, A. & Shor, J. *DoubletDetection* 2018. http://doi.org/10.5281/zenodo.2678042.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

83. Pedregosa, F. *et al.* Scikit-learn: machine learning in Python. *Journal of Machine Learning Research* (2011).
84. Bishop, C. M. *Pattern Recoginiton and Machine Learning* (2006).
85. Lun, A. T. L. *et al.* EmptyDrops: distinguishing cells from empty droplets in droplet-based single-cell RNA sequencing data. *Genome Biology* **20,** 63 (2019).
86. Lai, L., Alaverdi, N., Maltais, L. & Morse, H. C. Immunophenotyping mouse cell surface antigens: Nomenclature and immunophenotyping. *The Journal of Immunology* (1998).
87. Watts, C. Capture and processing of exogenous antigens for presentation on MHC molecules. *Annual Review of Immunology* **15,** 821–850 (1997).
88. Uchida, J. *et al.* Mouse CD20 expression and function. *International Immunology* (2004).
89. Hünig, T., Beyersdorf, N. & Kerkau, T. CD28 co-stimulation in T-cell homeostasis: a recent perspective. *ImmunoTargets and Therapy* **4,** 111 (2015).
90. Filion, L. G., Izaguirre, C. A., Garber, G. E., Huebsh, L. & Aye, M. T. Detection of surface and cytoplasmic CD4 on blood monocytes from normal and HIV-1 infected individuals. *Journal of Immunological Methods* **135,** 59–69 (1990).
91. Zhou, Z., Ye, C., Wang, J. & Zhang, N. R. Surface protein imputation from single cell transcriptomes by deep neural networks. *Nature Communications* **11,** 1–10 (2020).
92. Traag, V., Waltman, L. & van Eck, N. J. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports* **9** (2018).
93. Kirchner, J. & Bevan, M. J. ITM2A is induced during thymocyte selection and T cell activation and causes downregulation of CD8 when overexpressed in CD4+CD8+ double positive thymocytes. *Journal of Experimental Medicine* **190,** 217–228 (1999).
94. Macias-Garcia, A. *et al.* Ikaros and B1 cells Ikaros is a negative regulator of B1 cell development and function. *Journal of Biological Chemistry* (2016).
95. Shi, Z. *et al.* Human CD8+ CXCR3+ T cells have the same function as murine CD8+ CD122+ Treg. *European Journal of Immunology* **39,** 2106–2119 (2009).
96. Liu, J., Chen, D., Nie, G. D. & Dai, Z. CD8+CD122+ T-cells: A newly emerging regulator with central memory cell phenotypes. *Frontiers in Immunology* **6** (2015).
97. Merad, M., Sathe, P., Helft, J., Miller, J. & Mortha, A. The Dendritic cell lineage: ontogeny and function of dendritic cells and their subsets in the steady state and the inflamed setting. *Annual Review of Immunology* **31,** 563–604 (2013).
98. Durai, V. & Murphy, K. M. Functions of murine dendritic cells. *Immunity* **45,** 719–736 (2016).
99. Scott, R. E., Ghule, P. N., Stein, J. L. & Stein, G. S. Cell cycle gene expression networks discovered using systems biology: Significance in carcinogenesis. *Journal of Cellular Physiology* **230,** 2533–2542 (2015).
100. Doty, R. T. *et al.* Coordinate expression of heme and globin is essential for effective erythropoiesis. *Journal of Clinical Investigation* **125,** 4681–4691 (2015).
101. Sagar *et al.* Deciphering the regulatory landscape of $\gamma\delta$ T Cell development by single-cell RNA-sequencing. *bioRxiv.* https://doi.org/10.1101/478529 (2018).
102. Burmeister, Y. *et al.* ICOS controls the pool size of effector-memory and regulatory T cells. *The Journal of Immunology* **180,** 774–782 (2008).
103. Chen, X. & Oppenheim, J. J. Resolving the identity myth: key markers of functional CD4 + FoxP3 + regulatory T cells. *International Immunopharmacology* (2011).
104. Sunderkötter, C. *et al.* Subpopulations of mouse blood monocytes differ in maturation stage and inflammatory response. *The Journal of Immunology* **172,** 4410–4417 (2004).
105. Marcovecchio, P. M. *et al.* Scavenger receptor CD36 directs nonclassical monocyte patrolling along the endothelium during early atherogenesis. *Arteriosclerosis, thrombosis, and vascular biology* **37,** 2043–2052 (2017).
106. Miller, J. C. *et al.* Deciphering the transcriptional network of the dendritic cell lineage. *Nature Immunology* (2012).
107. Borges Da Silva, H. *et al.* Splenic macrophage subsets and their function during blood-borne infections. *Frontiers in Immunology* **6,** 480 (2015).
108. Tardif, M. R. *et al.* Secretion of S100A8, S100A9, and S100A12 by neutrophils involves reactive oxygen species and potassium efflux. *Journal of Immunology Research* (2015).

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

109. Fehniger, T. A. *et al.* Acquisition of Murine NK cell cytotoxicity requires the translation of a pre-existing pool of Granzyme B and Perforin mRNAs. *Immunity* **26,** 798–811 (2007).

110. Bendelac, A., Rivera, M. N., Park, S.-H. & Roark, J. H. Mouse CD1-Specific NK1 T Cells: Development, Specificity, and Function. *Annual Review of Immunology* **15,** 535–562 (1997).

111. Zhang, J. *et al.* Characterization of Siglec-H as a novel endocytic receptor expressed on murine plasmacytoid dendritic cell precursors. *Blood* **107,** 3600–3608 (2006).

112. Sawai, C. M. *et al.* Transcription factor Runx2 controls the development and migration of plasmacytoid dendritic cells. *Journal of Experimental Medicine* **210,** 2151–2159 (2013).

113. Castro, C. D. & Flajnik, M. F. Putting J Chain back on the map: How might its expression define plasma cell development? *The Journal of Immunology* **193,** 3248–3255 (2014).

114. Dutta, P. *et al.* Macrophages retain hematopoietic stem cells in the spleen via VCAM-1. *Journal of Experimental Medicine* **212,** 497–512 (2015).

115. Hobeika, E., Dautzenberg, M., Levit-Zerdoun, E., Pelanda, R. & Reth, M. Conditional selection of B cells in mice with an inducible B cell development. *Frontiers in Immunology* **9,** 1806 (2018).

116. Benjamini, Y. & Hochberg, Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)* **57,** 289–300 (1995).

117. Burda, Y., Grosse, R. & Salakhutdinov, R. *Importance weighted Autoencoders* in *International Conference on Learning Representations* (2016).

118. Liberzon, A. *et al.* Databases and ontologies Molecular signatures database (MSigDB) 3.0. *Bioinformatics* **27,** 1739–1740 (2011).

119. Blei, D. M. & Lafferty, J. D. A correlated topic model of Science. *The Annals of Applied Statistics* **1,** 17–35 (2007).

120. Ioffe, S. & Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* (2015).

121. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* **15,** 1929–1958 (2014).

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

5 Supplementary Figures

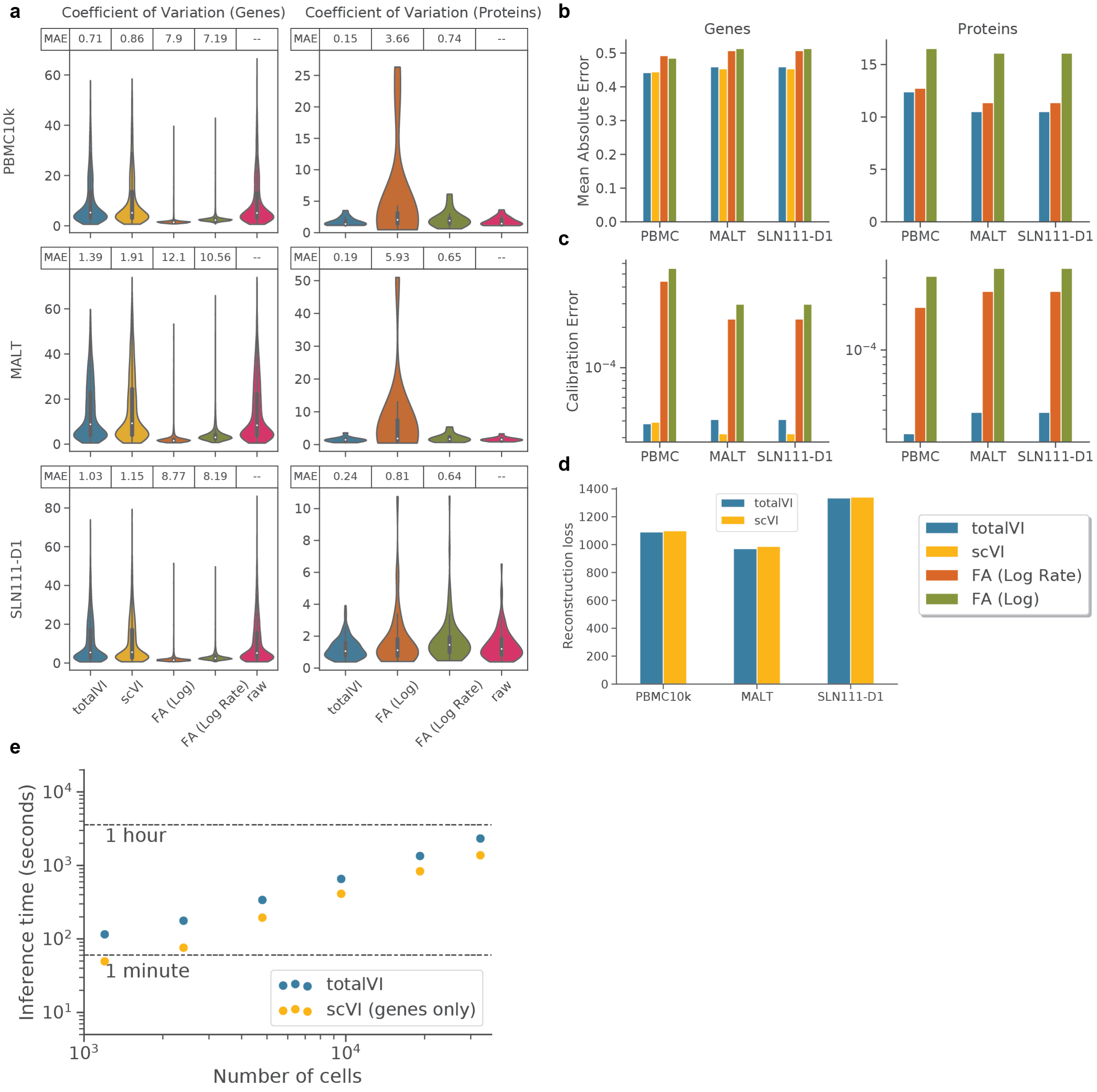

Figure S1: Evalutation of totalVI model. **a**, Posterior predictive check of coefficient of variation (CV) of genes and proteins. For each of the PBMC10k, MALT, and SLN111-D1 datasets and for each model (totalVI, scVI, factor analysis with normalized input) the average coefficient of variation from posterior predictive samples was computed for each feature. Violin plots summarize the distribution of CVs for genes and proteins. Mean absolute error (MAE) between raw data CVs and average posterior predictive CV are reported. **b**, MAE between held out data and posterior predictive mean separated by genes and proteins for each model and dataset. **c**, Calibration error of held-out data reported separately for genes and proteins. **d**, Held-out reconstruction loss of genes for scVI and totalVI. **e**, Inference time for totalVI and scVI across cells randomly subsampled to different levels from SLN-all. scVI was run with only genes.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

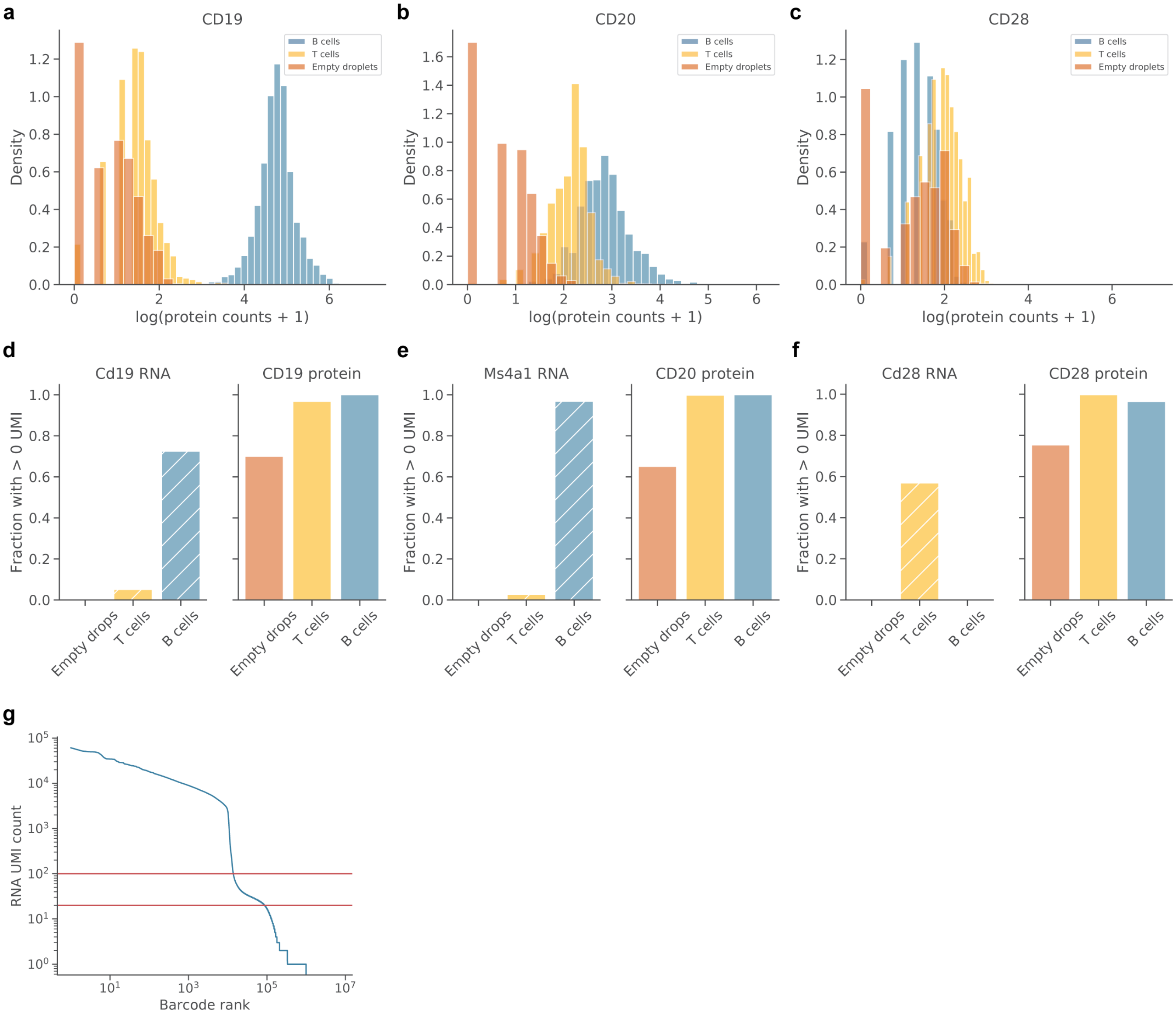

Figure S2: Protein background in cells and empty droplets. a-c, Histogram of log(protein counts + 1) in the SLN111-D1 dataset for B cells, T cells, and empty droplets (Methods 4.8) for CD19 **(a)**, CD20 **(b)**, and CD28 **(c)**. **d-f**, Fraction of empty droplets, B cells, or T cells with > 0 UMIs detected for a given RNA (left, hatched) or protein (right, solid). RNA/proteins displayed are Cd19/CD19 **(d)**, Ms4a1/CD20 **(e)**, and Cd28/CD28 **(f)**. **g**, Barcode rank plot for all barcodes detected in the SLN111-D1 dataset. Red lines at 20 and 100 RNA UMI counts indicate the lower and upper bounds, respectively, used to define empty droplets in (a-f).

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

a

Cell type	Protein	totalVI ROC AUC	GMM ROC AUC
B cells	CD19	**0.999868**	0.999759
B cells	CD45R-B220	**0.999927**	0.997479
B cells	CD20	**0.999567**	0.860060
B cells	I-A-I-E	**0.994174**	0.972592
T cells	CD5	**0.999854**	0.997426
T cells	TCRbchain	**0.999975**	0.997695
T cells	CD90.2	**0.999988**	0.998642
T cells	CD28	**0.999931**	0.832834
CD4 T cells	CD4	**0.999959**	0.996998
CD8 T cells	CD8a	**0.999973**	0.998501
CD8 T cells	CD8b	**0.999847**	0.998083

Classification of cell types by marker proteins

b

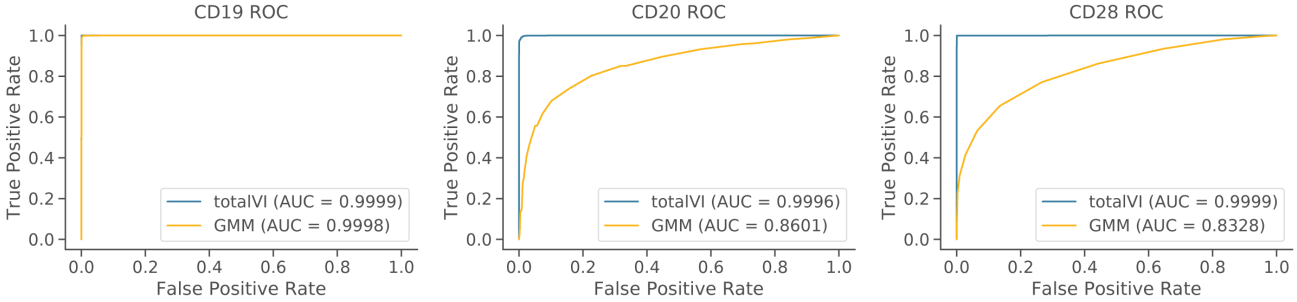

Figure S3: Classification of cell types by marker proteins. a, Performance of totalVI and a Gaussian mixture model (GMM) fit on all cells for each protein of the SLN111-D1 dataset. Area under the receiver operating characteristic curve (ROC AUC score) was calculated using as input either the totalVI foreground probability or GMM foreground probability where the indicated cell type was the positive population out of all B and T cells. Bolded ROC AUC scores indicate higher values (better performance). Highlighted in blue are two proteins for which totalVI noticeably outperformed the GMM. **b**, Receiver operating characteristic (ROC) curves for CD19 (B cells), CD20 (B cells), or CD28 (T cells).

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

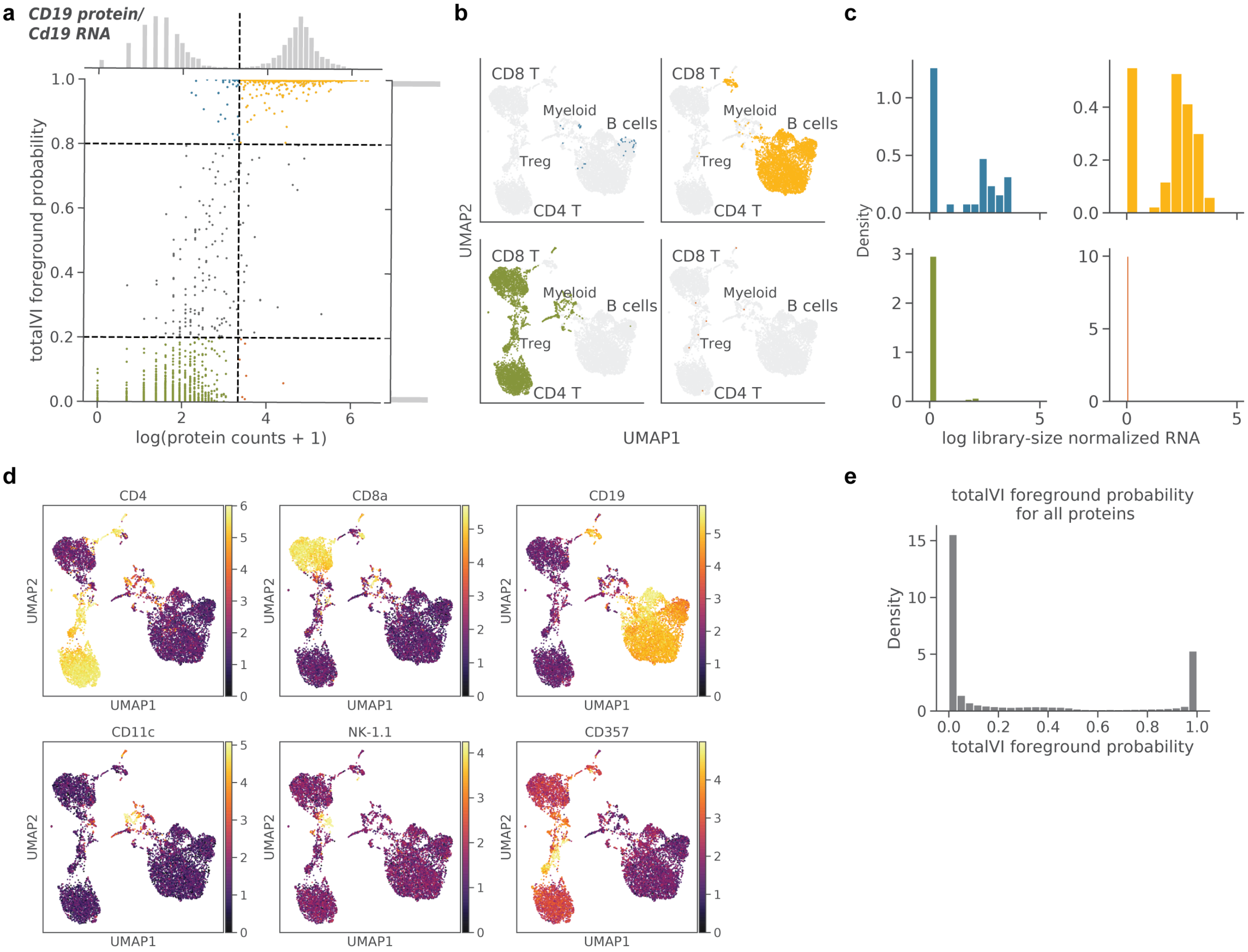

Figure S4: totalVI decouples protein foreground and background. totalVI was applied to the SLN111-D1 dataset. **a-c**, CD19 protein (encoded by Cd19 RNA). **(a)** totalVI foreground probability vs log(protein counts + 1). Vertical line denotes protein foreground/background cutoff determined by a GMM. Horizontal lines denote foreground probability of 0.2 and 0.8. Cells with foreground probability greater than 0.8 or less than 0.2 are colored by quadrant, while the remaining cells are gray. **(b)** UMAP plots of the totalVI latent space. Each quadrant contains cells from the corresponding quadrant of (a) in color with the remaining cells in gray. **(c)** RNA expression (log library-size normalized; Methods 4.8) for cells colored in (a). **d**, UMAP plots of the totalVI latent space colored by (log(counts + 1)) of cell type marker proteins (Table 4). **e**, totalVI foreground probability for all proteins across all cells in the SLN111-D1 dataset.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

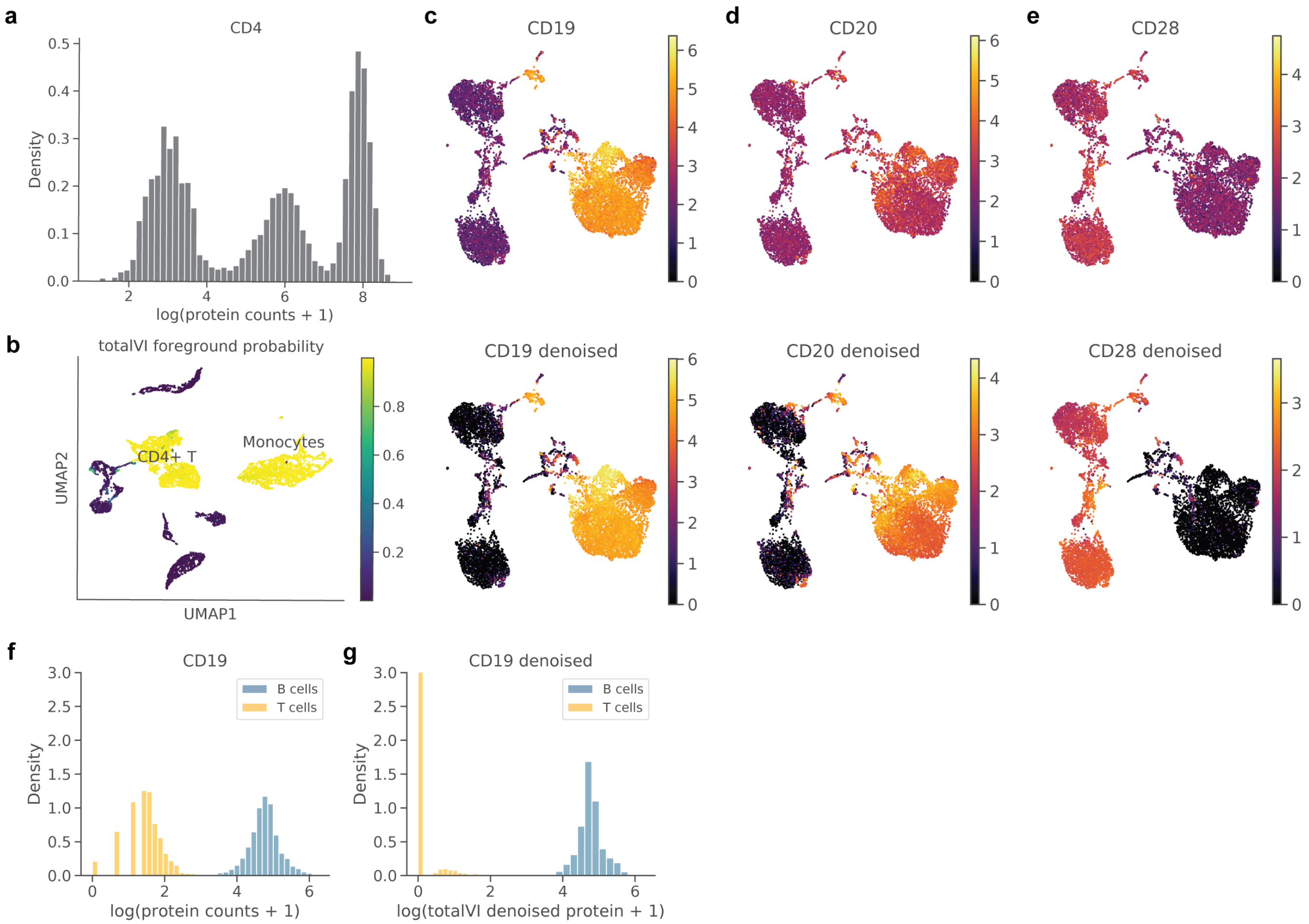

Figure S5: totalVI decouples foreground and background for trimodal protein distributions and denoises protein data. a, b CD4 protein expression in the PBMC10k dataset. **(a)** Trimodal distribution of log(protein counts + 1). **(b)** UMAP plot of the totalVI latent space colored by totalVI foreground probability. **c-e**, UMAP plots of the totalVI latent space for the SLN111-D1 dataset. Plots are colored by log(protein counts + 1) (top) and log(totalVI denoised protein + 1) (bottom) for CD19 **(c)**, CD20 **(d)**, and CD28 **(e)**. **f, g**, Distributions of log(protein counts + 1) **(f)** and log(totalVI denoised protein + 1) **(g)** for CD19 protein in B and T cells. y-axis is truncated at 3.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

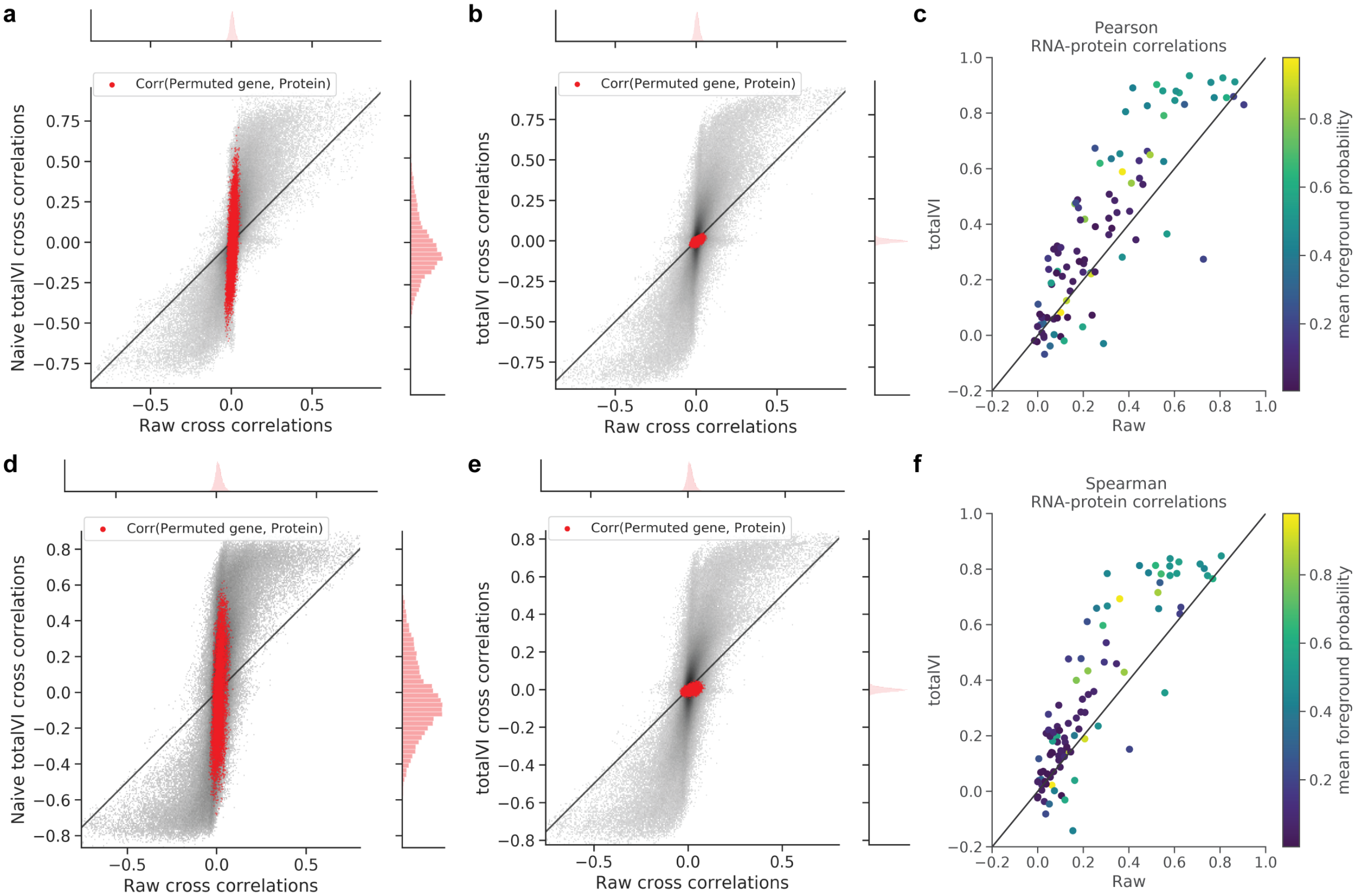

Figure S6: RNA-protein correlations. a, b, 2d density plots of Pearson correlations between all RNA and protein features in the SLN111-D1 dataset as well as 100 additional genes whose expression was randomly permuted. Correlations between all proteins and the randomly permuted genes are colored in red. Raw correlations were calculated between log library-size normalized RNA and log(protein counts + 1). **(a)**, Naive totalVI correlations were calculated between totalVI denoised RNA and totalVI denoised proteins. **(b)**, totalVI correlations were calculated between denoised RNA and proteins sampled from the posterior (Methods 4.3). **c**, Pearson correlations between each protein and its encoding RNA for all proteins with a unique encoding RNA, colored by the mean probability foreground of the protein across all cells. totalVI correlations were calculated as in (b) and raw correlation were calculated as in (a, b). **d-f**, Same as (a-c), but for Spearman correlations.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

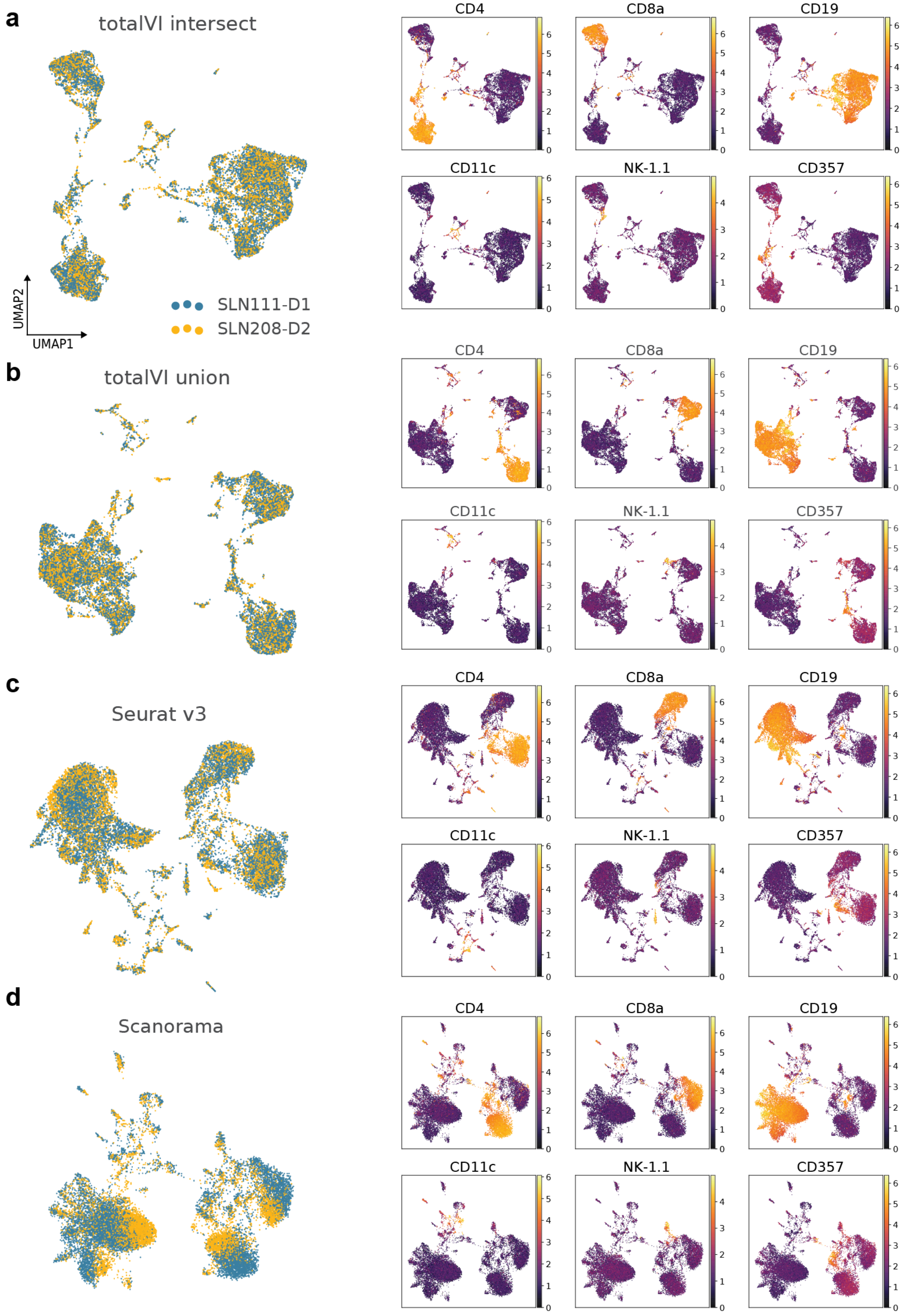

Figure S7: UMAP embeddings of integration methods on spleen and lymph node data. a-d, For each method, UMAP plots colored by dataset, and by log(counts + 1) of key marker proteins (Table 4).

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

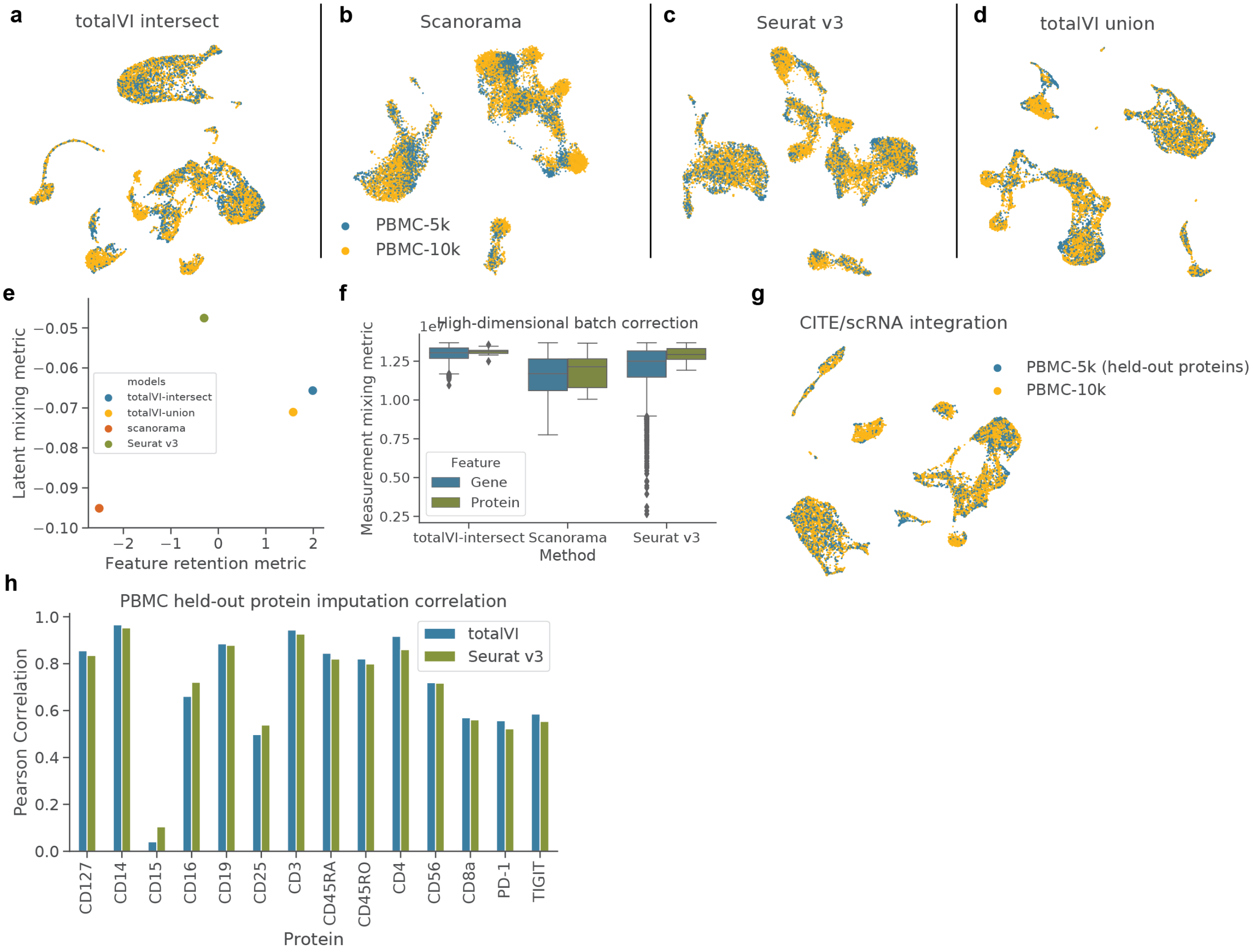

Figure S8: Benchmarking of integration methods on PBMC data. Integration methods were applied to PBMC10k and PBMC5k. **a-d**, UMAP plots of integrated latent spaces. **e**, Latent mixing metric and feature retention metric for each method. **f**, Measurement mixing metric applied indivdually to each batch-corrected feature. **g**, UMAP plot of integrated latent space from totalVI union mode when holding out the proteins from PBMC5k. **h**, Pearson correlation between imputed and observed proteins (log scale) from PBMC5k for totalVI and Seurat v3.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

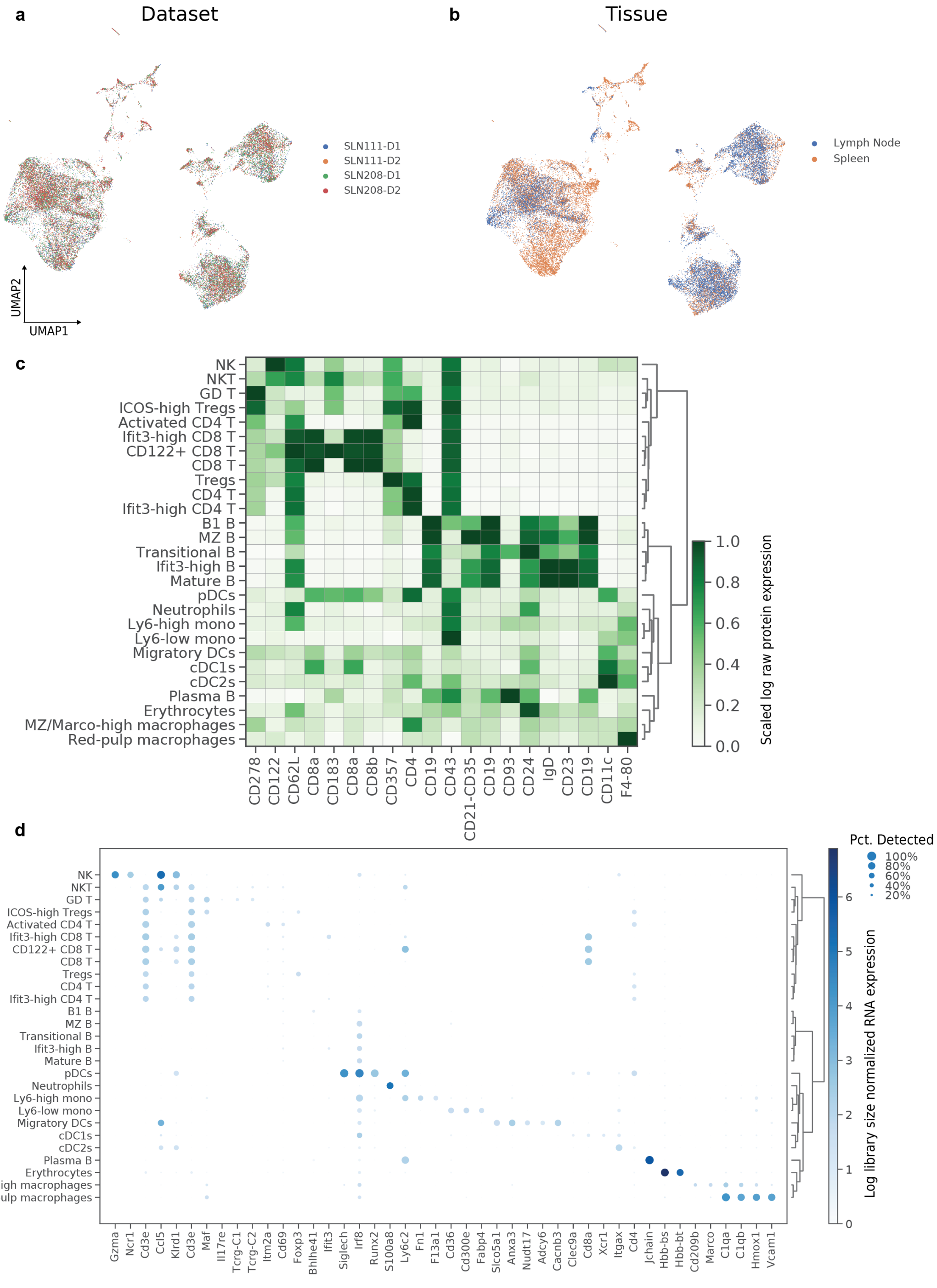

Figure S9: Integration of SLN-all with totalVI-intersect. **a, b**, UMAP plot of SLN-all colored by **(a)** dataset, and **(b)** tissue. **c**, Heatmap of proteins used for annotation. Proteins (columns) are log(protein counts + 1) scaled by column for visualization. **d**, Dotplot of RNA markers used for annotation. RNA is log library size normalized.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

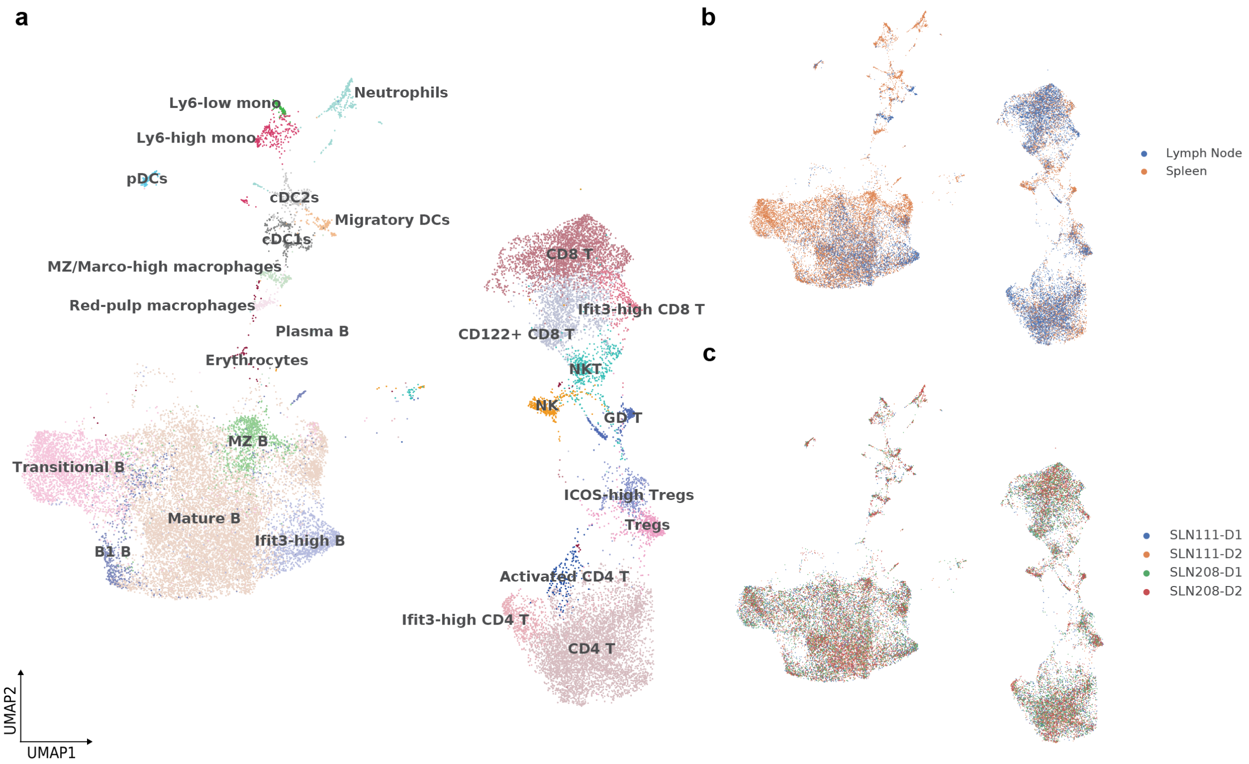

Figure S10: Integration of SLN-all with union of panels. **a-c**, UMAP plot of SLN-all colored by **(a)** cell types derived from manual annotation of model run with intersection of panels, **(b)** tissue, and **(c)**, dataset.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

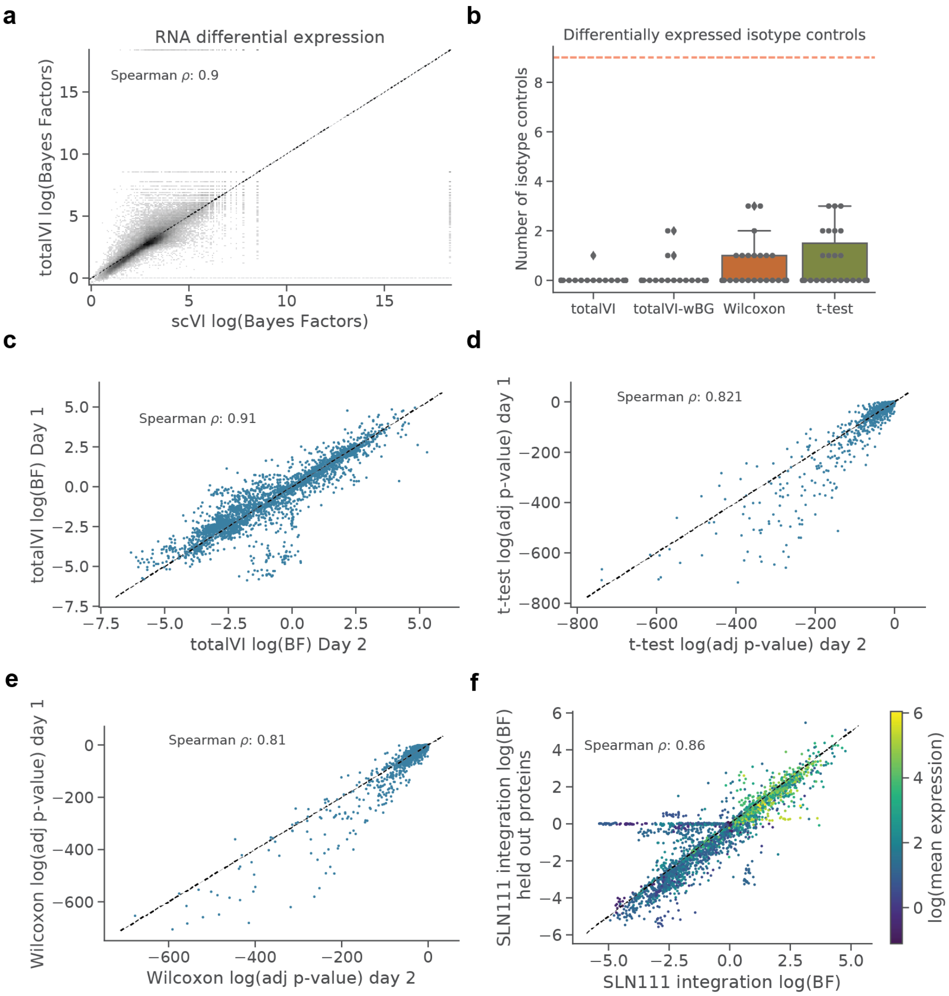

Figure S11: Differential expression analysis. **a**, 2d density plot of totalVI and scVI log Bayes factors for genes. Bayes factors were computed for each gene in one-vs-all tests on the SLN111-D1 dataset. **b**, Number of isotype controls called differentially expressed in one-vs-all tests for totalVI, totalVI-wBG (totalVI test without background removal), Wilcoxon rank-sum, and t-test. Tests were applied to SLN208-D1, for which isotype controls were retained. **c-e**, Significance level (Bayes factors for totalVI, adjusted *p*-value for frequentist tests) for proteins in one-vs-all tests computed on SLN111-D1 and SLN111-D2 for each of **(c)** totalVI, **(d)** t-test, **(e)** Wilcoxon. **f**, Bayes factors for proteins in one-vs-all tests computed on the SLN111 datasets integrated with and without the SLN111-D2 proteins held-out. Differential expression tests for both model fits were conditioned on SLN111-D1. Bayes factors are colored by the average protein expression from SLN111-D1.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

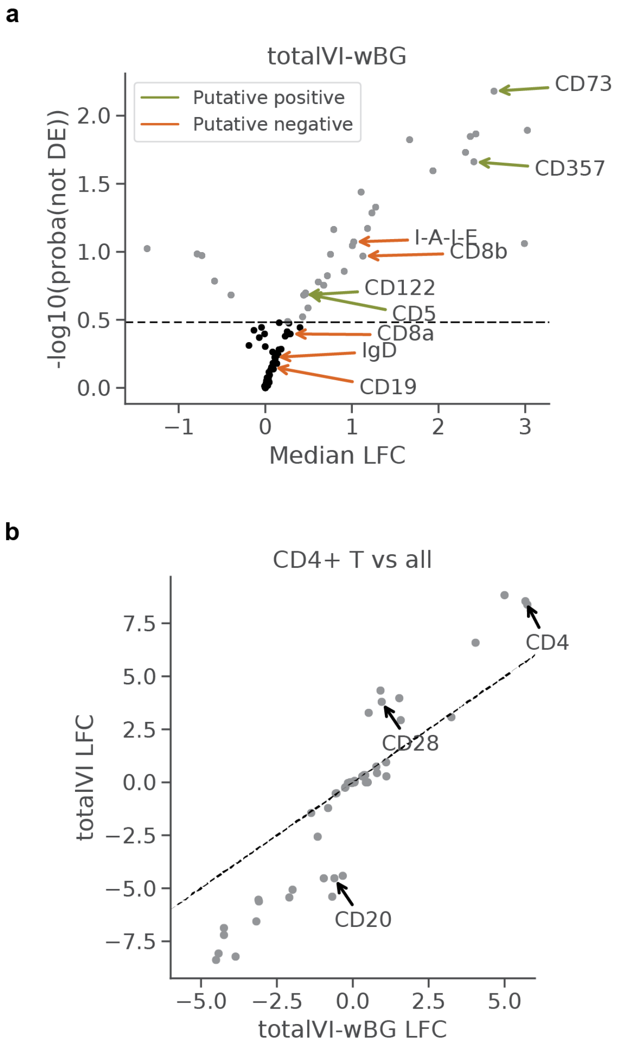

Figure S12: Differential expression using totalVI-wBG, which does not correct for the protein background component. a Volcano plot for the ICOS-high Tregs vs CD4 T cells test. The same putative positives and negatives as Figure 4d are highlighted here. **b** The LFC estimates for totalVI and totalVI-wBG on the CD4 T cells versus all others test.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

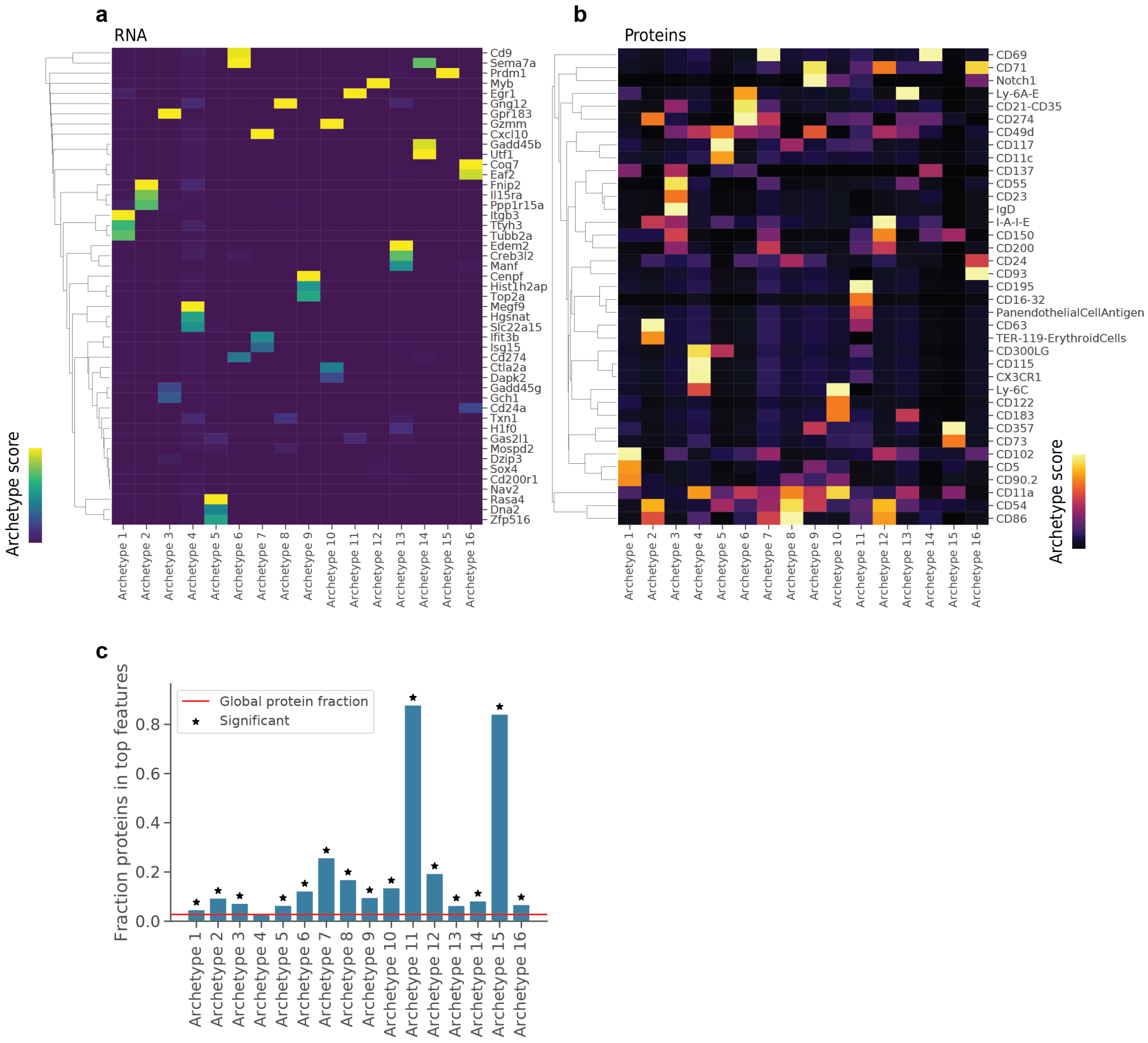

Figure S13: Interpreting totalVI latent dimensions with archetypal analysis. a, b, Heatmap of top **(a)** gene and **(b)** protein features for each archetype. The archetype score corresponds to the standard scaled archetypal expression profiles (Methods 4.13). Heatmaps are individually column normalized for visualization. **c**, Fraction of proteins in top archetypal features for each archetype. Features in each archetype were selected in the "top" if they had an archetype score of greater than 2. For these features, we performed a hypergeometric test to determine if proteins were over-represented in this feature set relative to the global distribution of feature types.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

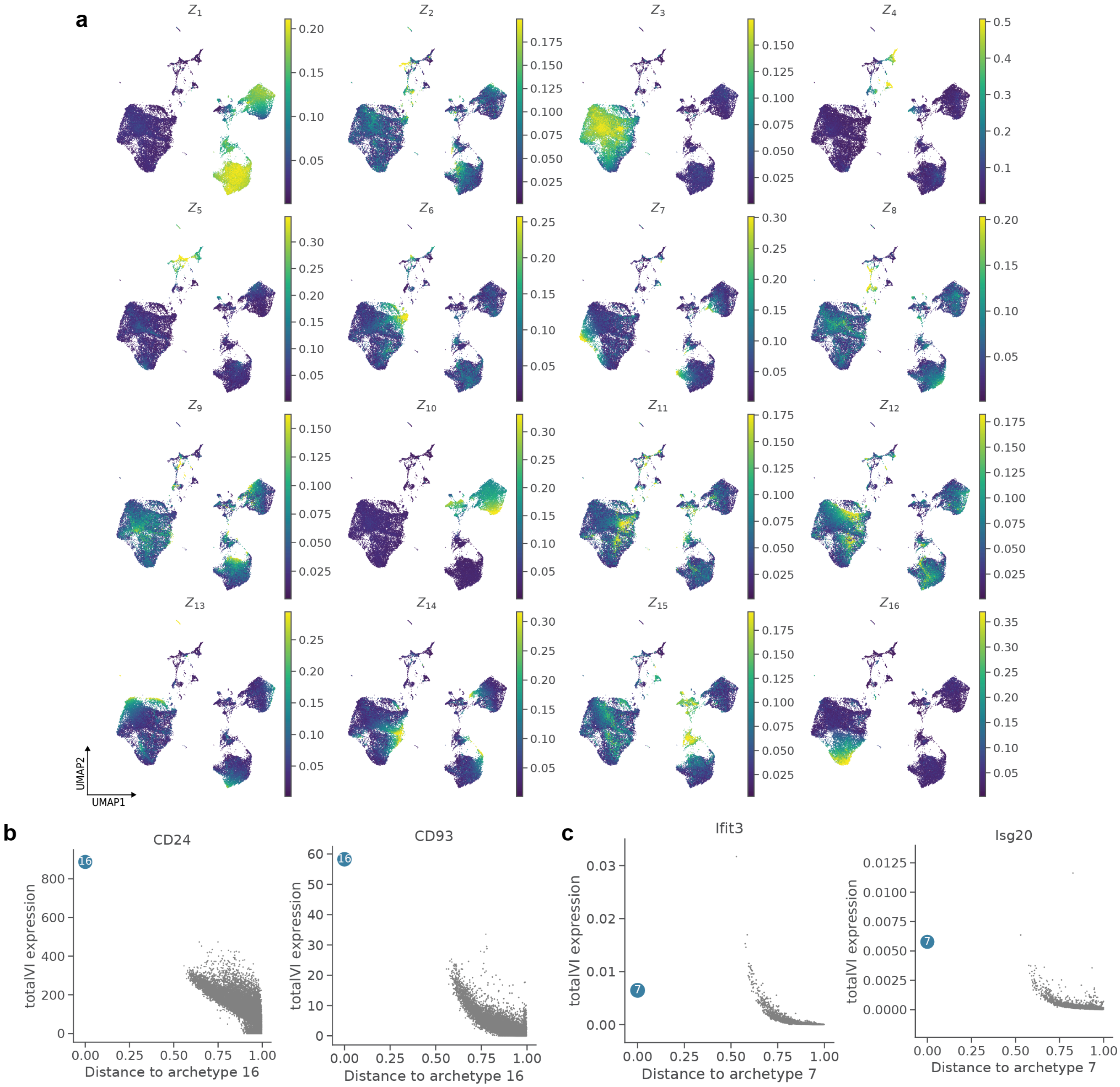

Figure S14: Visualization of archetypes in totalVI-intersect model of SLN-all. **a**, UMAP plots of SLN-all cells colored by latent dimension value. **b**, totalVI protein expression for CD24 and CD93 proteins as a function of distance to archetype 16. **c**, totalVI denoised expression for Isg20 and Ifit3 genes as a function of distance to archetype 7. Archetype is colored in blue, all other cells in grey.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

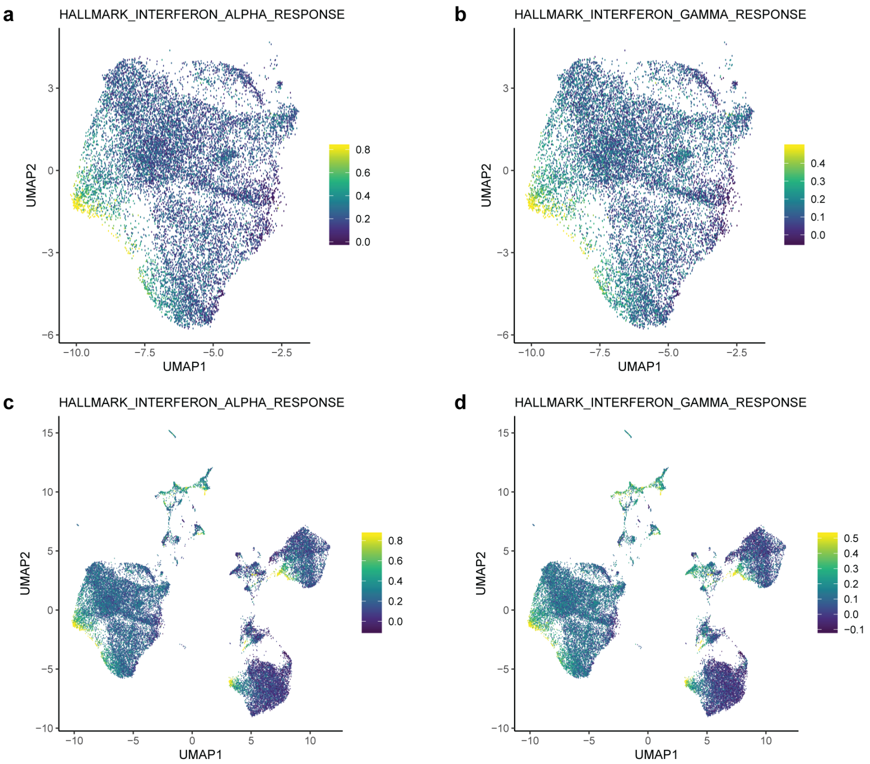

Figure S15: Interferon signatures in the mouse spleen and lymph node. a, b, UMAP of totalVI latent space for B cells of the SLN-all dataset **(a)** colored by the Hallmark Interferon Alpha Response signature score and **b** colored by the Hallmark Interferon Gamma Response signature score (Methods 4.14). **c, d**, Same as in (a, b), but for all cells in SLN-all.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

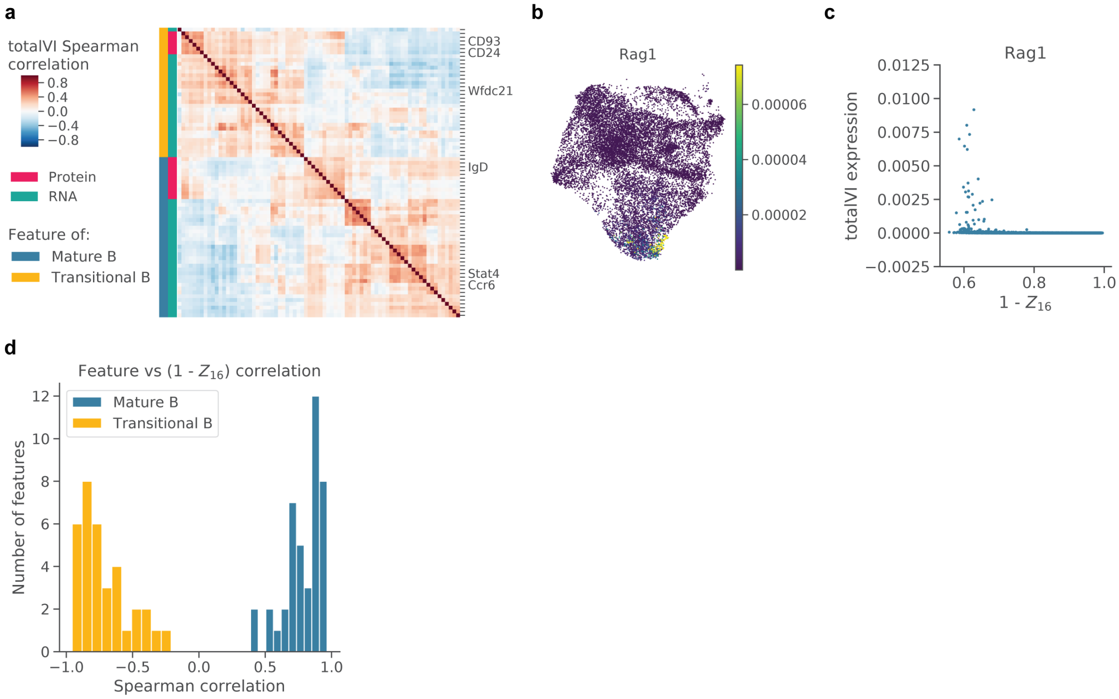

Figure S16: totalVI identifies correlated modules of RNA and proteins that are associated with the maturation of transitional B cells. **a**, totalVI Spearman correlations in mature B cells between the same RNA and proteins as in Figure 5h. Features were hierarchically clustered within mature B cells. **b**, UMAP of the totalVI latent space colored by totalVI RNA expression of Rag1. **c**, totalVI RNA expression of Rag1 as a function of 1 - Z_{16} (the totalVI latent dimension associated with transitional B cells). **d**, Histogram of Spearman correlations between each feature in (a) and 1 - Z_{16}.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

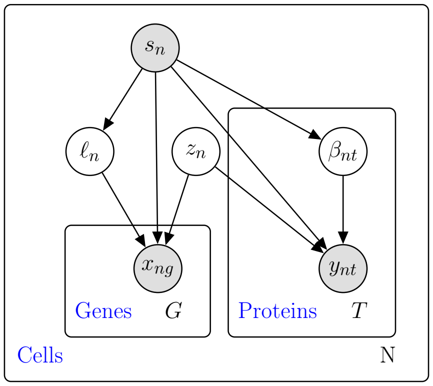

Figure S17: totalVI probabilistic graphical model. Shaded nodes represent observed random variables. Unshaded nodes represent latent variables. Edges denote conditional independence. Rectangles ("plates") represent independent replication.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

A Protein considerations

A.1 Experimental considerations

Sources of technical variation in CITE-seq experiments, particularly protein background, are dependent on the experimental method itself. There are a number of potential experimental sources of background. We primarily discussed ambient antibodies and non-specific antibody binding. Another potential source of background could arise from oligonucleotide barcodes that become dissociated from their conjugated antibody. Similarly to barcoded antibodies, ambient oligonucleotide barcodes could contaminate cell-containing droplets or could non-specifically bind to the cell surface. In this study, we do not distinguish between background due to antibodies or background due to free oligonucleotide barcodes.

Although in our experiments we used the standard CITE-seq protocol, there are a number of protocol modifications that could change the amount of background. For instance, increasing the number of washes after staining cells with antibodies could reduce ambient background. Alternatively, a buffer modification could reduce the amount of non-specific binding. Both washing and blocking are frequently considered in flow cytometry protocol designs. However, implementing these protocol changes in an effort to eliminate background could come with trade-offs; reducing background by washing and blocking would likely reduce true signal by reducing total cell numbers or blocking specific binding, respectively.

Another common experimental practice to modulate the amount of background is antibody titration, meaning that different antibodies are added to the experiment at different concentrations. At the optimal titration, an antibody would have the maximal signal-to-noise ratio. This would require the antibody to be present at a sufficient concentration to specifically bind its target protein and generate a detectable signal, but not at so high a concentration as to increase protein background by binding non-specifically or remaining at high concentration in the ambient solution. In a CITE-seq experiment, it is possible that the recommended antibody concentration is too low to detect foreground signal from a given protein. Finding the optimal concentration for each antibody might be challenging, since the optimal concentration might be different for different cell types or experimental systems. If titrations are modified per antibody, there are a few points to consider. When antibodies are titrated at different concentrations, it becomes infeasible to quantify absolute protein levels. For example, it would not be possible to determine if one protein was expressed at a higher level than another protein. In addition, even if every protein in the cell were measured with a theoretical unbiased antibody panel, the sum of all protein counts from a cell (referred to as the protein library size) could not serve as a meaningful estimate of total protein molecules in a cell or cell size because the relative amounts of each protein have been manipulated.

In our modeling and analysis, we considered whether protein library size should be taken into account. In scRNA-seq data, we consider library size to be a nuisance factor that is reflective of a combination of sequencing depth and cell size. In CITE-seq experiments where only selected markers are measured, there is no guarantee that the markers selected are representative of the total protein content in the cell. For example, a cell with few detected protein counts might in reality express other unmeasured proteins at higher levels, meaning this cell's total counts reflect the selection of markers rather than reflecting nuisance variation like sequencing depth or cell size. Therefore, treating protein library size as a nuisance factor does not make sense in this context. Because measured proteins are biased in this manner, we do not assume that the protein data is compositional, as is assumed by other methods that use a centered log ratio transformation for normalization. In the future, more unbiased protein panels might necessitate further consideration of protein library size in CITE-seq data analysis.

For proteins that are expressed at low levels or are only expressed in rare cell types, it might be necessary to increase sequencing depth to increase the sensitivity to detect these molecules. In addition, sequencing depth could play a role in the ability of totalVI to decouple protein foreground and background (i.e., with more counts, it might become easier to separate what appear to be overlapping foreground and background distributions). Determining the optimal sequencing depth for protein panels could be an important cost consideration in CITE-seq experiments, particularly as the size of protein panels increases. Since in the CITE-seq protocol RNA and protein libraries are prepared independently, future work could determine the value of these two molecules in various downstream analysis tasks to make recommendations for sequencing depth for each library.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Because the barcoded antibodies used in this study came from clones that have been previously validated, we were surprised to find that some common protein markers (e.g., IgM, CCR7) appeared to have little or no signal. Aside from the consideration of titration and sequencing depth discussed above, an additional explanation could be the uniform staining conditions for all antibodies in the CITE-seq panel simultaneously. For example, the chemokine receptor CCR7 is a well-documented marker in T cells and typically requires staining at higher temperature and for longer times than other antibodies due to its constant cycling onto and off of the cell surface. For future CITE-seq experiments, it might be worthwhile to consider the optimal staining conditions (e.g., time and temperature) for each antibody independently rather than staining with all antibodies at once.

A.2 Modeling considerations

Guided by the points raised in Appendix A.1, we considered a variety of protein likelihoods before settling on the version used in this manuscript. Among our considerations were the interpretability of the parameters as well as how well the likelihood captured our view on the CITE-seq protein data generating process. For example, we considered models that included a latent variable for the protein library size (analogous to ℓ_n for RNA), though as discussed in Appendix A.1, protein library size and RNA library size do not convey the same information.

We also considered alternative models to decouple protein background. Initially, we attempted to use models that assume every cell receives the same distribution of protein background scaled by some cell-specific scalar. However, we found these models inadequate for decoupling the protein signal, which again suggests that ambient antibodies can not fully explain the protein background.

The likelihood we used in this manuscript,

$$y_{nt} \mid z_n, \beta_n, s_n \sim \text{NegativeBinomialMixture}(\beta_{nt}, \beta_{nt}\alpha_{nt}, \pi_{nt}), \tag{34}$$

also has some import downstream considerations. First, the mixture assumes that the observed counts for a given cell n and protein t are generated from either the background component (with probability π_{nt}) or the foreground component (with probability $1 - \pi_{nt}$). Despite the fact that the background mean parameter β_{nt} appears in the foreground mean $\beta_{nt}\alpha_{nt}$, this likelihood does not allow us to correct the foreground for possible background contamination. Here, the double usage of β_{nt} is to help identify the mixture model. In other words, we cannot "subtract the background" from y_{nt} that are determined to be in the foreground. Perhaps this limitation could be addressed in future work, in which different latent variables are associated with local and global sources of background; though this will require greater understanding of the experimental mechanisms previously discussed.

B Integrating out latent variables

Here we show that if

$$w \sim \text{Gamma}(\theta, \ell\rho) \tag{35}$$
$$x \mid w \sim \text{Poisson}(w) \tag{36}$$

then $x \sim \text{NegativeBinomial}(\ell\rho, \theta)$. Note that we have dropped all subscripts and each variable here is a scalar. While we parameterize the Gamma with its shape and mean, a more conventional form is

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

with its shape and rate, so $w \sim \text{Gamma}(\theta, \theta/\ell\rho)$

$$\begin{aligned} p(x) &= \int_0^\infty p(x \mid w)p(w)dw && (37)\\ &= \int_0^\infty \frac{w^x e^{-w}}{\Gamma(x+1)} \frac{(\theta/\ell\rho)^\theta}{\Gamma(\theta)} w^{\theta-1} e^{-\theta w/\ell\rho} dw && (38)\\ &= \frac{(\theta/\ell\rho)^\theta}{\Gamma(x+1)\Gamma(\theta)} \int_0^\infty w^{x+\theta-1} e^{-(1+\theta/\ell\rho)w} dw && (39)\\ &= \frac{(\theta/\ell\rho)^\theta}{\Gamma(x+1)\Gamma(\theta)} \frac{\Gamma(x+\theta)}{(1+\theta/\ell\rho)^{x+\theta}} && (40)\\ &= \frac{\Gamma(x+\theta)}{\Gamma(x+1)\Gamma(\theta)} \left(\frac{\theta/\ell\rho}{1+\theta/\ell\rho}\right)^\theta \left(\frac{1}{1+\theta/\ell\rho}\right)^x && (41)\\ &= \frac{\Gamma(x+\theta)}{\Gamma(x+1)\Gamma(\theta)} \left(\frac{\theta}{\ell\rho+\theta}\right)^\theta \left(\frac{\ell\rho}{\ell\rho+\theta}\right)^x && (42) \end{aligned}$$

In the fourth line, we use the fact that the integrand is an unnormalized gamma distribution. The final line is a negative binomial distribution with mean $\ell\rho$ and inverse dispersion θ. Therefore, we have a direct link between the parameters of the negative binomial and the underlying parameters of the Poisson rate. Finally, we note that we could have written $w \sim \text{Gamma}(\theta, \rho)$ and $x \mid w \sim \text{Poisson}(\ell w)$ and achieved the same result.

C totalVI implementation details

Evidence lower bound derivation Here we derive the Evidence lower bound (ELBO), which is ultimately used in optimizing the model and variational parameters. For shorthand, we drop subscripts and inference and generative parameters ν and η. The joint likelihood based on the totalVI generative model for a single cell factorizes as

$$p(x, y, \beta, z, \ell \mid s) = p(x \mid z, \ell, s)p(y \mid \beta, z, s)p(\beta \mid s)p(z)p(\ell \mid s). \quad (43)$$

In the model specification, we use the latent variable $z \sim \text{LogisticNormal}(0, I)$. Here we use the logistic normal definition of [119], in which a normal random variable $\delta \sim \text{Normal}(0, I)$ is transformed by a softmax function, embedding the random variable in the simplex. Thus, $z = \text{softmax}(\delta)$. However, the softmax function is not invertible, so for simplicity we consider the underlying latent variable δ. In this setting, z, which is ultimately the input to the decoder, is treated as a likelihood parameter. Therefore, we can rewrite the joint likelihood as

$$p(x, y, \beta, \delta, \ell \mid s) = p(x \mid \delta, \ell, s)p(y \mid \beta, \delta, s)p(\beta \mid s)p(\delta)p(\ell \mid s). \quad (44)$$

To perform variational inference, we define the variational posterior distribution as

$$q(\beta, \delta, \ell \mid x, y, s) = q(\beta \mid \delta, s)q(\delta \mid x, y, s)q(\ell \mid x, y, s). \quad (45)$$

The Evidence LOwer Bound (ELBO) is derived using Jensen's inequality. We use the shorthand notation $q(\beta, \delta, \ell) = q(\beta, \delta, \ell \mid x, y, s)$.

$$\begin{aligned} \log p(x, y \mid s) =& \log \mathbb{E}_{q(\beta,\delta,\ell)} \left[\frac{p(x, y, \beta, \delta, \ell \mid s)}{q(\beta, \delta, \ell)}\right] && (46)\\ \geq& \mathbb{E}_{q(\beta,\delta,\ell)} \left[\log \frac{p(x, y, \beta, \delta, \ell \mid s)}{q(\beta, \delta, \ell)}\right] && (47)\\ =& \mathbb{E}_{q(\beta,\delta,\ell)} \left[\log p(x, y \mid \beta, \delta, \ell, s)\right] + \mathbb{E}_{q(\beta,\delta,\ell)} \left[\log \frac{p(\beta \mid s)p(\delta)p(\ell \mid s)}{q(\beta, \delta, \ell)}\right] && (48)\\ =& \mathbb{E}_{q(\beta,\delta,\ell)} \left[\log p(x, y \mid \beta, \delta, \ell, s)\right] - \text{KL}\left(q(\ell) \parallel p(\ell \mid s)\right) - \text{KL}\left(q(\delta) \parallel p(\delta)\right) && (49)\\ &- \mathbb{E}_{q(\delta)} \left[\text{KL}\left(q(\beta) \parallel p(\beta \mid s)\right)\right] && (50) \end{aligned}$$

To compute the KL divergences of lognormal random variables we note that the KL divergence is invariant to invertible transformations, so the KL can be computed in closed form using the KL

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Algorithm 2: Inference for totalVI

```
Initialize inverse dispersion parameters $\theta, \phi$, background parameters $c_t, d_t$ and encoder/decoder
  neural network parameters
for iteration $i = 1, 2, ...,$ do
    Randomly choose $M$ cells for mini-batch $\mathcal{C}$
    for each cell $n$ in $\mathcal{C}$ do
        Encode $x_n, y_n, s_n$ to obtain approximate posterior parameters
        Sample $z_n, \ell_n, \beta_n$ from approximate posterior $q(\beta_n, \delta_n, \ell_n \mid x_n, y_n, s_n)$
        Decode $z_n, s_n$ to obtain likelihood parameters $\alpha_n, \pi_n, \rho_n$
        for each gene $g$ do
            Compute $\log p(x_{ng} \mid \ell_n, z_n, s_n)$
        for each protein $t$ do
            Compute $\log p(y_{nt} \mid \beta_{nt}, z_n, s_n)$
        Update parameters using gradient of ELBO estimate
```

between normal random variables. The log likelihood of a negative binomial mixture distribution is computed using numerically stable functions in Pytorch (Appendix D). The ELBO derived here is amenable to the reparameterization trick used to train VAEs [25]. Estimates of the expectations in the ELBO are taken via Monte Carlo and noisy gradients of the ELBO are used in a stochastic optimization scheme. A sketch of the inference procedure for totalVI is in Algorithm 2.

Approximate posterior specification The approximate posterior distributions are specified by neural networks. In particular, one neural network takes as input the triple (x, y, s) and outputs the parameters of $q(\delta \mid x, y, s)q(\ell \mid x, y, s)$. An additional neural network maps (δ, s) to the mean and variance parameters of $q(\beta \mid z, s)$ through z. The variational distributions match their priors in family (e.g., $q(\delta \mid x, y, s)$ is a Gaussian with diagonal covariance matrix). A posterior draw of z, which we used as input to clustering and visualization algorithms, as well as used for archetypal analysis is then obtained by

1. Draw δ from $q(\delta \mid x, y, s)$
2. Set $z = \text{softmax}(\delta)$

Neural networks The encoder neural network has one shared hidden layer of 256 nodes followed by a layer of 512 nodes. The output of the 512 nodes are split in half and are used as input for linear layers that parameterize $q(\delta \mid x, y, s)$ and $q(\ell \mid x, y, s)$, respectively. The final encoder neural network of one hidden layer and 256 hidden nodes takes as input (z, s) and outputs the parameters of $q(\beta \mid \delta, s)$. The parameters of δ are 20-dimensional mean and variance parameters. The decoder consists of three individual neural networks each with one hidden layer and 256 nodes. The first maps to the parameters of the mean of the RNA likelihood (ρ_n). The second maps to the foreground mean of the protein likelihood (α_n). The third maps to the mixing parameter of the protein likelihood mixture (π_n). Each of these decoder networks takes as input (z, s). Furthermore, (z, s) are reinjected at each hidden layer. All neural networks use batch normalization [120], dropout regularization [121], and ReLU activations in hidden layers. The model parameters θ, ϕ, c, and d are treated as global neural network parameters, optimized to maximize the ELBO.

Hyperparameters The neural network architecture previously described was used throughout this manuscript without modification. There are a number of other hyperparameters used to train neural networks that we also held constant in all experiments. This includes the learning rate of the optimizer (`lr=4e-3`), the size of the training test (90%), the KL warmup scheme (0.75×`NumCells` minibatches, or 213 epochs with the fixed training set size of 90%), and the number of training epochs (500 epochs). An early stopping scheme is performed with respect to the 10% of cells in the test set. If there is no improvement of the held-out ELBO of the test set with 30 epochs, the learning rate is multiplied by 0.6. If there is no improvement after 45 epochs, the inference procedure is stopped. The totalVI package includes a module for likelihood-guided hyperparameter tuning, which can be used to revisit the default parameters, especially as the experimental protocol evolves.

bioRxiv preprint doi: https://doi.org/10.1101/2020.05.08.083337; this version posted May 12, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

D Numerical considerations for the negative binomial mixture distribution

For a single negative binomial mixture component, we use numerically stable functions provided by PyTorch (e.g., the log gamma function). For a mixture of negative binomials, we rewrite the distribution to use numerically stable functions like `logsumexp` and `softplus`.

Let $p^b(y) = p(y \mid z, \mu, s, v = 1)$ be the probability mass function for the background and $p^f(y) = p(y \mid z, \mu, s, v = 0)$ be the probability mass function for the foreground. Then by integrating over v (recalling that $v \sim \text{Bernoulli}(\pi)$),

$$\log p(y \mid z, \mu, s) = \log\big(\pi p(y \mid z, \mu, s, v = 1) + (1 - \pi) p(y \mid z, \mu, s, v = 0)\big). \tag{51}$$

We now rewrite this in a form more amenable for optimization. We recall from Algorithm 1 that $\pi = h_\pi(z, s; \Omega)$. Thus, with $c(z, s) = \text{logit}(h_\pi(z, s))$, then $\pi = 1/(1 + \exp(-c(z, s)))$. Also, let $\mathcal{S}$ be the softplus function: $x \to \log(1 + e^x)$.

$$\begin{aligned}
\log p(y \mid z, \mu, s) &= \log\big(\pi p^b(y) + (1 - \pi) p^f(y)\big) && (52)\\
&= \log\big(p^f(y) + \pi p^b(y) - \pi p^f(y)\big) && (53)\\
&= \log\left(\frac{p^f(y) + p^f(y)e^{-c(z,s)}}{1 + e^{-c(z,s)}} + \frac{p^b(y) - p^f(y)}{1 + e^{-c(z,s)}}\right) && (54)\\
&= \log\left(p^b(y) + p^f(y)e^{-c(z,s)}\right) - \log\left(1 + e^{-c(z,s)}\right) && (55)\\
&= \log\left(e^{\log p^b(y)} + e^{\log p^f(y)} e^{-c(z,s)}\right) - \log\left(1 + e^{-c(z,s)}\right) && (56)\\
&= \text{logsumexp}\left(\log p^b(y), \log p^f(y) - c(z, s)\right) - \mathcal{S}(-c(z, s)) && (57)
\end{aligned}$$