Data-driven optimization of biomarker panels in highly multiplexed imaging

Huangqingbo Sun[1], **Jiayi Li**[1], **and Robert F Murphy**[1,✉]

[1]Computational Biology Department, Carnegie Mellon University, Pittsburgh, PA, 15213

Motivation: Multiplexed protein imaging methods provide valuable information on complex tissue structure and cellular heterogeneity. However, costs increase and image quality decreases with the number of biomarkers imaged, and the number of markers that can be measured in the same tissue sample is limited.

Results: In this work, we propose an efficient algorithm to choose a minimal predictive subset of markers that for the first time allows the prediction of full images for a much larger set of markers. We demonstrate that our approach also outperforms previous methods for predicting cell-level marker composition. We also demonstrate that an effective minimal subset can be selected even if the desired full set is too large to be imaged on the same samples.

Availability: All code and intermediate results are available in a Reproducible Research Archive at `https://github.com/murphygroup/CODEXPanelOptimization`.

Multiplex imaging | Machine learning | Protein biomarker | Computational imaging

Correspondence: *murphy@cmu.edu*

Introduction

Multiplexed imaging methods enable researchers to analyze individual cell properties and their spatial relationships in complex tissues. These methods, like co-detection by indexing (CODEX), allow imaging of many dozens of markers in the same tissue sample (1). However, unlike spatial transcriptomics, in which many thousands of different gene transcipts can be measured, spatial proteomics allows only a small fraction of all proteins to be imaged in the same sample.

Deep learning has been widely used to solve a variety of biomedical image analysis problems including cell segmentation (2–5), super-resolution microscopy enhancement (6, 7), and cell type identification (8–10). In particular, it helps to learn complex relationships between different signal sources from large collections of images. A particularly useful approach for imaging is '*in silico* labeling', in which unmeasured signals are predicted from easily acquired common reference signals. It includes predicting fluorescent tags of subcellular components from unlabeled microscope images (11–14), virtual histological staining of tissue images (15–17), and predicting immunofluorescence or directly inferring cell types from immunohistochemical stained images (18, 19). This concept can be extended to predict a large number of biomarkers from images measuring a smaller number. For example, (20) described a method to select 7 informative markers out of 40 markers that would enable accurate prediction of cell types, and showed the effectiveness of the approach by performing 7-plex imaging (21). Their method began by converting multiplexed images to biomarker levels in each cell, and then learned to predict marker intensities at the single-cell level from the 7 measured marker intensities as well as cell morphology features learned separately by another machine learning model.

In this work, we introduce a new heuristic method for iteratively selecting a set of biomarkers that allow accurate prediction of full images. In contrast to the method of (20), we consider the problem of marker selection from an optimization standpoint, and our method focuses on selecting biomarkers by explicitly modeling the spatial relationship between marker intensities and uses a neural network to directly infer the multiplexed fluorescence image instead of predicting expression at the single-cell level. Instead of selecting a unified panel to enable cell type prediction across multiple tissues, our approach can produce biomarker panels optimized for specific tissues. Our results demonstrate that panels selected by our method outperform those produced by (20). In addition, we consider a more difficult setting in which all markers of interest cannot be imaged simultaneously. In this case, only subsets of the markers can be imaged in the same sample, meaning that deep learning models cannot be trained directly as some pairs of markers cannot be co-detected. We show that our proposed method can effectively address this problem, and the method can be easily extended to a very large number of markers of interest.

Methods

In this section, we introduce the mathematical framework of the problem and the proposed heuristic approach. Formally, we denote biomarkers of interest as a set of random variables (RVs) $\mathcal{X} = \{X_1, \ldots, X_N\}$, and our goal is to partition $\mathcal{X}$ into $\mathcal{X}_{\text{in}}$ and $\mathcal{X}_{\text{out}} \subseteq \mathcal{X} \backslash \mathcal{X}_{\text{in}}$, where $\mathcal{X}_{\text{in}}$ refers to the minimal input set and $\mathcal{X}_{\text{out}}$ refers to the prediction set. We denote the partition function as $\delta: \mathcal{X} \to \{0, 1\}$, let δ_i denote $\delta(X_i)$. Then, we have $\mathcal{X}_{\text{in}} = \{X_i : \delta_i = 1\}$. With the partition, the next step is to construct a predictive model $f \in \mathcal{F}: \mathcal{X}_{\text{in}} \to \mathcal{X}_{\text{out}}$ using some learning algorithm. In this work, we use an empirical risk minimization (ERM) approach to train f. We denote $\mathcal{D}_{\text{in}}$ as the distributions over $\mathcal{X}_{\text{in}}$. With the choice of loss used in the learning algorithm, L, we define the population risk $\mathcal{R} = \mathbb{E}_{\mathcal{D}_{\text{in}}}[L(f)]$. Then, the problem of finding a minimal in-

DRAFT

put set can be viewed as an optimization problem,

$$\underset{\mathcal{X}_{\text{in}}}{\operatorname{argmin}} |\mathcal{X}_{\text{in}}| \text{ s.t. } \rho, \tag{1}$$

where ρ refers to the chosen stopping criterion.
In general, finding an exactly minimal set of biomarkers is computationally hard.

Approximation algorithms.

Basic version. Given the hardness of the problem, we propose a heuristic algorithm here. Our algorithm starts with an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where nodes in $\mathcal{V}$ are associated with the RVs in $\mathcal{X}$. We use nodes $v_i \in \mathcal{V}$ and RV X_i interchangeably in the following text. $\mathcal{E}$ denotes the edge set of $\mathcal{G}$ where $\mathcal{E} = \{(X_i, X_j) : \forall i, j \in [|\mathcal{X}|], i \neq j\}$. Our approach is to weight the edges in $\mathcal{G}$ by the dissimilarity between the RVs associated with the nodes and iteratively update the edge weights by the unpredictability remaining after training a new model. Assume we have a training set S and a validation set P. Initially, edges are weighted by

$$w_{ij} = \mathbb{E}_{S_{ij}}[\xi(X_i, X_j)], \ (X_i, X_j) \in \mathcal{E}, \tag{2}$$

where S_{ij} denotes the marginal empirical distribution with respect to $X_i \times X_j$ over S, and ξ refers to some dissimilarity measure over $\mathcal{X} \times \mathcal{X}$. After initializing $\mathcal{G}$, nodes in $\mathcal{V}$ are weighted by a map $q : \mathcal{V} \to \mathbb{R}$, where $q(v)$ denotes the weight of node v, $\forall v \in \mathcal{V}$. For a node whose associated RV is not in $\mathcal{X}_{\text{in}}$, the node weight is the minimal weight of activated edges it connects, and 0 for a node whose associated RV is in $\mathcal{X}_{\text{in}}$ (Line 10 of Algorithm 1). Here, an activated edge means it connects one node whose associated RV is in $\mathcal{X}_{\text{in}}$ and the other's associated node in $\mathcal{X}_{\text{out}}$. Our goal is to have node weights measuring the unpredictability of their associated RVs and edge weights measuring the loss occurring when predicting the value of one connected node from the other (observed) connected node. The algorithm aims to gradually reduce the overall unpredictability among all markers of interest by iteratively including informative markers into $\mathcal{X}_{\text{in}}$ and refitting the model using the updated set of input markers. Iteratively adding new RVs to $\mathcal{X}_{\text{in}}$ results in monotonically decreasing risk if we further assume the model is 'properly' trained (typically true). For models at t-th and $(t+1)$-th round, f_t and f_{t+1}, the RVs in the input and output sets of f and f_{t+1} are denoted as $\mathcal{X}_{\text{in}}^{(t)}$, $\mathcal{X}_{\text{out}}^{(t)}$, $\mathcal{X}_{\text{in}}^{(t+1)}$, and $\mathcal{X}_{\text{out}}^{(t+1)}$, respectively, with $\mathcal{X}_{\text{in}}^{(t)} \subseteq \mathcal{X}_{\text{in}}^{(t+1)}$ and $\mathcal{X}_{\text{out}}^{(t+1)} \subseteq \mathcal{X}_{\text{out}}^{(t)}$, $\mathcal{R}_P^{(t+1)} \leq \mathcal{R}_P^{(t)}$. Our method is presented formally in Algorithm 1.
In practice, we started with $\mathcal{X}_{\text{in}}$ containing only one RV, that whose associated node had the minimal node weight where edge weights were calculated by eq. 2 using L1-norm for ξ. Then, in each round, the model was fit using ERM and predictions were made for RVs in $\mathcal{X}_{\text{out}}$. Then, the weights of activated edges were updated using the model's prediction performance on the validation set, and weights of edges connecting nodes both in $\mathcal{X}_{\text{in}}$ were set to 0 (Line 9 in Algorithm 1). Also, the node weights of RVs in $\mathcal{X}_{\text{in}}$ were set to 0 (as they

Algorithm 1 Minimal input set selection (`MISS`)

Require: S, P, $\mathcal{F}$, $\mathcal{X}$, ρ
1: $t \leftarrow 0$
2: $\mathcal{X}_{\text{in}} \leftarrow \emptyset$
3: setup edge weights by eq. 2
4: $\mathcal{X}_{\text{in}} \leftarrow \mathcal{X}_{\text{in}} \cup \operatorname{argmin}_{X_i} \deg(v_i)$ ▷ deg refers to node degree, include RV with minimal node degree
5: $\mathcal{X}_{\text{out}} \leftarrow \mathcal{X} \backslash \mathcal{X}_{\text{in}}$
6: **while** not ρ **do**
7: $t \leftarrow t+1$
8: $f_t = \operatorname{argmin}_{f \in \mathcal{F}} \mathcal{R}_S$ ▷ train the model by ERM
9: $\forall X_i, X_j \in \mathcal{X}, \ X_i \neq X_j,$

$$w_{ij} \leftarrow \begin{cases} \mathbb{E}_{P_j}[L(f_t)] & X_i \in \mathcal{X}_{\text{in}}, \ X_j \in \mathcal{X}_{\text{out}} \\ 0 & X_i, X_j \in \mathcal{X}_{\text{in}} \end{cases}$$

▷ reassign edge weights
10:

$$\forall X_i \in \mathcal{X}, \ q(v_i) \leftarrow \begin{cases} \min_{X_j} w_{ij}, \text{ s.t. } \delta_j = 1 & X_i \in \mathcal{X}_{\text{out}} \\ 0 & X_i \in \mathcal{X}_{\text{in}} \end{cases}$$

▷ reassign node weights
11: $X \leftarrow$ `expectedImprovement`$(\mathcal{X}_{\text{in}}, \mathcal{X}_{\text{out}}, q, \mathcal{W})$
12: $\mathcal{X}_{\text{in}} \leftarrow \mathcal{X}_{\text{in}} \cup X$
13: $\mathcal{X}_{\text{out}} \leftarrow \mathcal{X} \backslash \mathcal{X}_{\text{in}}$
14: **end while**
15: **return** $\mathcal{X}_{\text{in}}$

Algorithm 2 subroutine: expectedImprovement

Require: $\mathcal{X}_{\text{in}}$, $\mathcal{X}_{\text{out}}$, q, $\mathcal{W}$
1: $\mathcal{I} = \{i : X_i \in \mathcal{X}_{\text{out}}\}$
2: **for** each element $k \in \mathcal{I}$ **do**
3:

$$\forall X_i \in \mathcal{X}, \ q'(v_i) \leftarrow \begin{cases} \min_{X_j} w_{ij}, \\ \text{s.t. } \delta_j = 1 \text{ or } j = k & X_i \in \mathcal{X}_{\text{out}} \\ 0 & X_i \in \mathcal{X}_{\text{in}} \text{ or } i = k \end{cases}$$

4: $\psi(k) \leftarrow \sum_{i=1}^{|\mathcal{X}|} q'(v_i)$
5: **end for**
6: $p \leftarrow \operatorname{argmin}_{k \in \mathcal{I}} \psi(k)$
7: **return** X_p

were observed, they are perfectly reliable). Node weights of RVs in $\mathcal{X}_{\text{out}}$ were set to the minimal edge weight among connecting edges to nodes in $\mathcal{X}_{\text{in}}$ (reflecting the assumption that the RV associated with the unobserved node can be predicted from an observed RV). Note that in Line 9 of Algorithm 1, for an unobserved RV, $X_j \in \mathcal{X}_{\text{out}}$, the edge weights are the same for any edges connecting this unobserved X_j with another observed RV. This is because we assume f here is a single model that takes all RVs in $\mathcal{X}_{\text{in}}$. However, if f refers to an ensemble of models, say $f = \{f_{ij} : X_i \in \mathcal{X}_{\text{in}}, X_j \in \mathcal{X}_{\text{out}}\}$ we expect $w_{ij} = \mathbb{E}_{P_j}[L(f_{t,ij})]$. Also, in Line 10 of Algorithm

DRAFT

1, when f is a single model, for an unobserved node, the weights will be the same for edges between it and other observed nodes, so taking the minimum does not matter. However, if the edge weights are not the same, we expect to take minimal edge weight here.

The principle of 'expectedImprovement' is that we select an RV in $\mathcal{X}_{\text{out}}$ and add it to $\mathcal{X}_{\text{in}}$ in order to decrease the risk the most. In practice, we selected the RV that would most decrease the summation of the node weights, $\sum_{i\in[|\mathcal{X}|]} q(v_i)$.

Distributed version. The basic version of the algorithm assumes the training and validation sets consist of images with all biomarkers of interest observed. However, this is not always true when the number of biomarkers of interest is large. For example, suppose we have 200 biomarkers of interest, but current imaging technology only allows us to have up to 60 markers in a single image. Then, the basic version of our algorithm will fail as it requires the samples in the S and P to contain all RVs in $\mathcal{X}$ in order to initialize $\mathcal{G}$. We, therefore, extended the algorithm to solve this problem as presented in Algorithm 3.

For simplicity, we first consider the case that the biomarkers of interest, $\mathcal{X}$, can be measured using only two panels. We denote the biomarkers in the two panels by $\mathcal{X}_A$ and $\mathcal{X}_B$ respectively, where $\mathcal{X}_A \cup \mathcal{X}_B = \mathcal{X}$. Here, we assume the sizes of both panels are less than or equal to the maximal number of markers that can be imaged simultaneously. Also, we expect there will be mutual markers appearing in both panels, namely $\mathcal{X}_M = \mathcal{X}_A \cap \mathcal{X}_B \neq \emptyset$. As illustrated in Figure 1 (b), for markers in each single panel, say $\{M_i : i = 1, \cdots, 6\}$ or $\{M_i : i = 1, 2, 9, 7, 8, 9\}$, we can first set up a subgraph using the same approach as for the single panel in the basic version described above. We then consider completing the whole graph by inferring the weights of edges between RVs in different panels. Recall that the edge weights are initialized by the dissimilarity of two RVs and updated by their unpredictability from one to the other RV associated with their connecting nodes. If we assume the loss (e.g., L1-norm) satisfies the Triangle inequality, for any two markers not in the same panel, say $X_i \in \mathcal{X}_A,\ X_i \notin \mathcal{X}_B$ and $X_j \in \mathcal{X}_B,\ X_j \notin \mathcal{X}_A$, we can get an uppperbound on the edge weight between X_i and X_j by $w_{ij} \leq w_{iq} + w_{qj}$, for $X_q \in \mathcal{X}_M$. For example, as illustrated in Figure 1 (b), we find the weight between nodes 4 and 7 by $w_{4,7} \leq \min(w_{1,4} + w_{1,4}, w_{2,7} + w_{2,7})$. By doing so, the distributed version degenerates to the basic version of the problem for which we already had an approximate solution above. Each round, two models are (re-)fit with respect to markers in $\mathcal{X}_A$ and $\mathcal{X}_B$ respectively using the training sets S_A and S_B; this requires the first marker in $\mathcal{X}_{\text{in}}$ to be chosen from $\mathcal{X}_M$. In principle, markers in $\mathcal{X}_M$ bridge the markers from different panels.

In practice, we first selected $\mathcal{X}_A$ out of $\mathcal{X}$ randomly with the size of panel A chosen to ensure the size of panel B is measurable. After measuring that set, and given a choice of a mutual set, say $|\mathcal{X}_M|$, we selected mutual markers to be included in panel B using Algorithm 1 from panel A. That is, $\mathcal{X}_B \subseteq \mathcal{X} \backslash \mathcal{X}_A \cup \mathcal{X}_M$. After measuring set B, we can interatively select a minimal predictive subset from $\mathcal{X}_A \cup \mathcal{X}_B$ as illustrated in Algorithm 3. Here, we retrained one or both of the models, f_{At} or f_{Bt}, during each round, depending on the panel where the last marker was selected. For the distributed version, the edge weights were initialized as follows

$$w_{ij} = \begin{cases} \mathbb{E}_{S^*_{ij}}[\xi(X_i, X_j)] & X_i, X_j \in \mathcal{X}_*,\ * = A, B \\ \frac{1}{2}\sum_{*=A,B} \mathbb{E}_{S^*_{ij}}[\xi(X_i, X_j)] & X_i,\ X_j \in \mathcal{X}_M \\ \min_{X_q \in \mathcal{X}_M} w_{iq} + w_{qj} & \text{otherwise} \end{cases} \tag{3}$$

When updating the edges, for the case $X_i \in \mathcal{X}_A \cap \mathcal{X}_{\text{in}},\ X_i \notin \mathcal{X}_M,\ \mathcal{Q} \subseteq \mathcal{X}_M \cap \mathcal{X}_{\text{in}},\ X_j \in \mathcal{X}_B$, we had $w_{ji} = \min_{X_q \in \mathcal{Q}}(w_{iq} + w_{qj})$. We present the distributed version in Algorithm 3. Note that the proposed algorithm is written for the case with 2 panels of biomarkers, but it is straightforward to extend to the case with multiple (≥ 2) panels.

Algorithm 3 Distributed version of `MISS`

Require: $\mathcal{F},\ L,\ \mathcal{X},\ \rho$

$\mathcal{X}_A \leftarrow$ `randomSelection`$(\mathcal{X})$ ▷ $|\mathcal{X}_A|$ is predetermined
acquire $S^A,\ P^A$,
$\mathcal{X}_M \leftarrow$ `MISS`$(S_A,\ P_A,\ \mathcal{F},\ \mathcal{X}_A,\ \tilde{\rho})$ ▷ $\tilde{\rho}$ is a dummy stop sign, $|\mathcal{X}_M|$ is predetermined
$\mathcal{X}_B \leftarrow (\mathcal{X} \backslash \mathcal{X}_A) \cup \mathcal{X}_M$
acquire S^B, P^B
$t \leftarrow 0$
$\mathcal{X}_{\text{in}} \leftarrow \emptyset$
setup edge weights by eq. 3
$X \leftarrow \text{argmin}_{X_i} \deg(v_i)$ s.t. $v_i \in \mathcal{V}_M$
$\mathcal{X}_{\text{in}} \leftarrow \mathcal{X}_{\text{in}} \cup X$
$\mathcal{X}_{\text{out}} \leftarrow \mathcal{X} \backslash \mathcal{X}_{\text{in}}$
while not ρ **do**
  $t \leftarrow t + 1$
  $f^A_t \leftarrow \text{argmin}_{f \in \mathcal{F}} \mathcal{R}_{S^A}$ $f^B_t \leftarrow \text{argmin}_{f \in \mathcal{F}} \mathcal{R}_{S^B}$
  $\mathcal{X}^*_{\text{in}} \leftarrow \mathcal{X}^* \cap \mathcal{X}_{\text{in}}, \mathcal{X}^*_{\text{out}} \leftarrow \mathcal{X}^* \cap \mathcal{X}_{\text{out}},\ * \in \{A, B, M\}$
  $\forall X_i, X_j \in \mathcal{X},\ X_i \neq X_j,$

$$w_{ij} \leftarrow \begin{cases} \mathbb{E}_{P^*_j}[L(f^*_t)] & X_i \in \mathcal{X}^*_{\text{in}},\ X_j \in \mathcal{X}^*_{\text{out}}, * = A, B \text{ or } M \\ 0 & X_i, X_j \in \mathcal{X}_{\text{in}} \\ \min_{X_q \in \mathcal{X}_M} w_{iq} + w_{qj} & \text{otherwise} \end{cases}$$

$$\triangleright\ \mathbb{E}_{P^M_j} = \mathbb{E}_{P^A_j \cup P^B_j},\ L(f^M_t) = L\left(\frac{f^A_t + f^B_t}{2}\right)$$

$$\forall X_i \in \mathcal{X},\ q(v_i) \leftarrow \begin{cases} \min_{X_j} w_{ij},\ \text{s.t. } \delta_j = 1 & X_i \in \mathcal{X}_{\text{out}} \\ 0 & X_i \in \mathcal{X}_{\text{in}} \end{cases}$$

  $X \leftarrow$ `expectedImprovement`$(\mathcal{X}_{\text{in}},\ \mathcal{X}_{\text{out}},\ q,\ \mathcal{W})$
  $\mathcal{X}_{\text{in}} \leftarrow \mathcal{X}_{\text{in}} \cup X$
  $\mathcal{X}_{\text{out}} \leftarrow \mathcal{X} \backslash \mathcal{X}_{\text{in}}$
end while
return $\mathcal{X}_{\text{in}}$

DRAFT

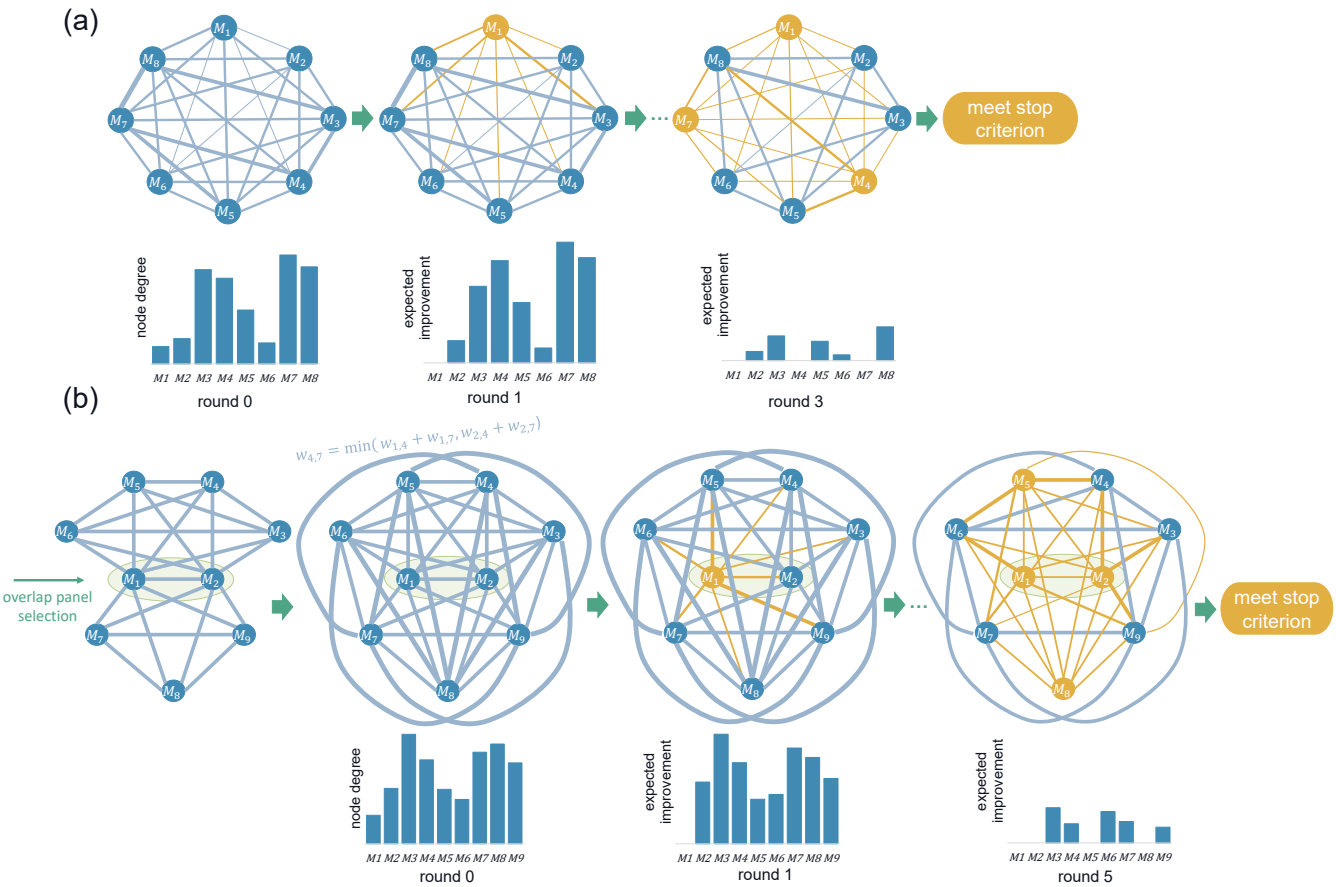

Fig. 1. Overview of our proposed Minimal Input Set Selection methods. (a) Basic version. An fully-connected RV graph is constructed with a node for each marker. In round 0, edge weights are calculated as the average pairwise L1-norm between marker intensities (shown in light blue). The thickness of the edge is shown proportional to the edge weight. We compute the node degree by the summation of weights of edges it is adjacent to (shown in the bar chart below the graph). The node with the minimal degree is selected as the first marker to be added to the minimal set (round 1). (The expected improvement for the other nodes is shown in the bar chart below the graph.) Then, RVs are iteratively added to the minimal set (shown in yellow), and edges attached to them are updated by the model predictive performance (also shown in yellow). During each round, we estimate how much the summation of node weights can be decreased by including an unobserved node into the minimal set (expected improvement) until we meet the stop criterion. (b) Distributed version. The set of markers is divided into two panels with some overlap (shown in light green). Models are then constructed from each panel separately. In Round 0, the graph is completed by assigning upperbound edge weights to edges connecting markers in different panels using the Triangle inequality. In Round 1, a marker in the overlapping set is selected (shown in yellow) and models are created for both panels using it as the input set. This allows estimation of the weights between the input node and other nodes. After updating the other edge weights, another node is added to the input set and the process repeated until the stop criterion is met.

Data preparation and machine learning. To illustrate the basic version of our algorithm, we used spleen and lymph node CODEX image datasets from data published by HuBMAP project (22). These contained 8 and 9 multichannel images of different tissue regions respectively. We split the datasets into training, validation, and test sets of 4:2:2 for the spleen dataset and 4:2:3 for the lymph node dataset. Each image in those data sets contains 29 measured biomarkers.

For illustrating Algorithm 3, we used HuBMAP datasets for large and small intestine image datasets containing 16 different multichannel images, where each image contains 46 measured biomarkers. We used these to select ("acquire") separate images of different panels assuming that the maximal number of markers that could be observed in a single panel was 26. We chose the number of markers to be shared between the two panels to be 6. For both datasets, 4 images were held out as a separate test set. The remaining 12 images, were split into 2 folds of 6 images. One fold was used as the training set for panel A as well as being used as the validation set for panel B, and vice versa. This simulates, each image sample only being imaged by markers in $\mathcal{X}_A$ or $\mathcal{X}_B$, but not both.

The intensities of each channel were normalized for each CODEX image using the same normalization method as (20) for spleen and lymph node datasets, as follows

$$x_i = \text{zscore}\left(\text{arcsinh}\left(\frac{x_i}{\max(5q_{0.2}(x_i), 15)}\right)\right), \quad \textbf{(4)}$$

where x_i denotes the i-th channel of the image, $q_{0.2}$ denotes the 20-th percentile, and $\text{zscore}(x) = \frac{x-\mu_x}{\sigma_x}$. For small and large intestine datasets, we directly applied zscore normalization, since the signal in these two datasets is relatively weak and sparse. Since the size of each CODEX image was very large, during training time, patches (channel number$\times 192 \times 192$) were randomly cropped from a CODEX image as the input or target of the model. However, during testing, the patches were cropped as a sliding window (not randomly) from a single CODEX image, and the original size image was recovered by stitching those patches.

We used convolutional neural networks as the predictive models in this work. In particular, the network architecture was a U-Net (23) with skip connections (24) which has been widely used in modern computer vision applications. The model was trained to minimize the empirical mean square error (MSE) using an Adam optimizer with a learning rate of 10^{-4}. The validation set was used to monitor the training process and the model with the lowest validation loss was selected.

As the minimal set selection required re-training models each round (Line 8 in Algorithm 1 and Line 14 in Algorithm 3), the model in round $t+1$, f_{t+1}, inherited the trained model weights from round t, f_t, for $t > 0$. Since the model architecture of f_{t+1} had an additional input channel and one fewer output channel compared to f_t, the weights of the selected channel in the input layer were randomly initialized in f_{t+1} and the weights of this channel in the output layer of f_t were not inherited. We randomly initialized the weights of f_1 following a normal distribution.

Evaluation metrics. Since the models were trained to minimize the empirical MSE, it was natural to first look at the model's performance on the test set in terms of MSE. However, pixel-level differences are not sufficient to reflect the quality of synthetic images since they are sensitive to noise in real images, but also hard to interpret in downstream applications. Therefore, we also sought to directly compare the cell type identification results from real and synthetic images. We segmented individual cells from test images, and calculated the average marker intensity for each cell. For spleen and lymph node datasets, we directly used segmented masks from HuBMAP. For intestine datasets, we segmented the cells using DeepCell Mesmer, a segmentation tool designed for large scale tissue images (2). Then, given a desired number of cell types, we performed KMeans clustering for cells from each normalized CODEX image to get cell type assignments for each cell. Cell type assignments were performed for real and synthetic images, respectively. We used the Rand Index (25) and normalized mutual information (26) to evaluate the agreement of cell type assignments between the real and synthetic images.

Results

Single panel experiment. We began by training models for image datasets of the spleen and lymph nodes. Each CODEX image was first normalized by eq. 4, and randomly cropped small patches were used to train neural networks to minimize the MSE between predicted marker intensity and observed intensity (see details in Methods). First, we compared the performance of models from our method and from (20) in terms of MSE reconstruction loss and Pearson correlation coefficients (PCC). Our method outperforms the 7-UP and 7-UP alternative on test MSE and PCC between predicted and real images, as shown in Figure 2 (a). Images synthesized by our model have lower MSE and higher PCC compared to other models. Then, we sought to recover the cell type assignments using the predicted images and compared them to the true cell type assignments from real images. Cell type assignments were done by KMeans clustering of single-cell average marker expression segmented from each CODEX image. We used Rand Index and normalized mutual information criteria (25, 26) to quantify the agreement between the cell type assignments on real and synthetic images with the same given cell type number. Three choices of the number of cell types were tested, providing coarse-to-fine cell type resolution. Each experiment was repeated 10 times to reduce the randomness in KMeans clustering. We performed a t-test on the resulting performances of each pair of methods and differences that are insignificant are indicated by an asterisk in Table 1 (using a p-value cutoff of 0.05). Overall, our method achieves better performance on both tissues. The results indicate that cell type assignment is relatively easier for the spleen dataset, as all methods work well and the margins of performances between different methods are small. Our selected minimal input set includes 7 markers for spleen images: CD5, DAPI, SMActin, CD31, CD15, Vimentin, and CD20. For the lymph node dataet, our method performs significantly better. The 7 markers chosen were FoxP3, CD3e, PanCK, CD163, HLA-DR, Vimentin, and CD8. For both datasets, our set is 4 markers different from the set selected by Wu.

Since the performance on the lymph node images was only moderate when using only 7 input channels, we continued training the lymph node model until the average Pearson correlation coefficients (PCCs) between predicted images and real images were greater than 0.86. This resulting in the choice of 12 input markers, except for the 7 markers reported above for lymph node images, 5 additional markers were: CD20, CD1c, LYVE-1, CD11c, and CD45RO. Using this set, the Rand Index and normalized mutual information of predicted cell types were 0.872 and 0.659 for 5 cell types, 0.908 and 0.675 for 10 cell types, and 0.926 and 0.677 for 15 cell types, respectively. The performances at each step in the training processes are shown in Figure 2 (a). This also illustrates the difference in difficulty for the two tissues. Figure 3 shows a comparison between randomly cropped patches of predicted and real images for the 3 markers that are predicted with the highest and lowest accuracies. It also shows the difference between the predicted and real images. The input markers (from $\mathcal{X}_{\text{in}}$) that are most similar in pattern to the predicted markers are also shown. When examined visually, all models trained on the selected minimal input sets were able to give reasonable predictions. We note that for the top 3 predicted markers, the colors were mostly close to black and white, indicating that all three markers express similar patterns. Also, as we check markers in the selected minimal set with the most similar expression patterns to those well-predicted markers, as shown in the bottom row of Figure 3, the predictions are similar to the patterns in the input set (but not exactly the same), and the models are able to infer those subtle differences. The underlying mode (group of similar patterns) of expression patterns coincides with our premise that biomarker expressions are predictable. In contrast, the inferior predictions show greater color variation,

DRAFT

Dataset	Spleen						Lymph node					
Number of cell types	5		10		15		5		10		15	
7-UP (20)	0.905*	**0.727**	0.896	0.660	0.924	0.676	0.807	0.542	0.854	0.526	0.892	0.552
7-UP alternative	0.899	0.714	0.911*	0.685	**0.935***	0.697*	0.845*	0.594	0.871	0.570	0.898	0.559
ours (7 markers)	0.905*	0.719	**0.917***	**0.700**	0.932*	**0.699***	0.845*	**0.606**	**0.878**	**0.585**	**0.900***	**0.564***

Table 1. We use Rand Index (left) and normalized mutual information (right) to measure the agreement of cell type assignments resulted from synthetic images to real images. We show the results of 7-UP and our proposed method for different cell type numbers. For each column, entries with asterisk symbols mean they are statistically insignificant tested by student t-test.

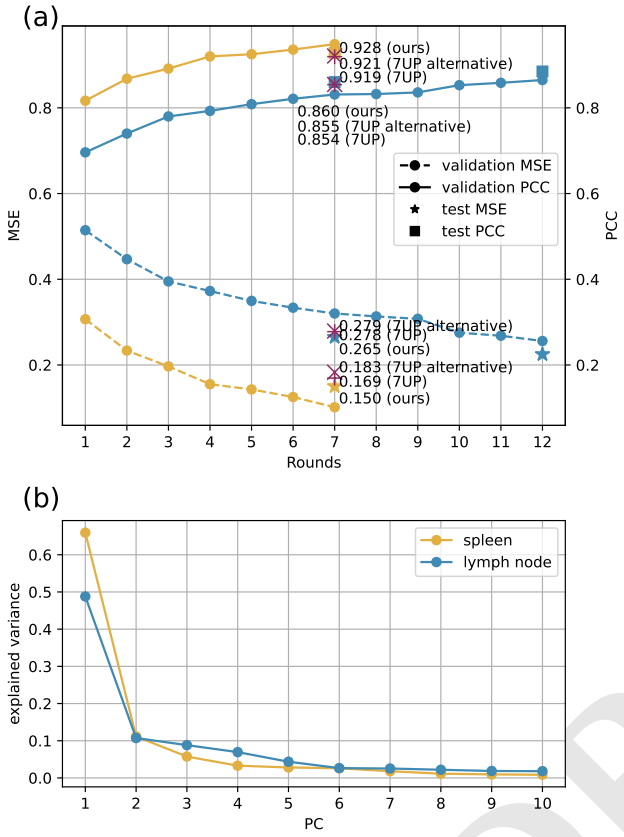

Fig. 2. (a) Prediction performance for iteratively chosen input sets. The curves marked with circles show the MSE and PCC between predicted and real images from the validation sets and the square and star show the values for predicted and real images from the test sets. For comparison with the 7-UP method, purple + and × symbols show the test MSE and PCCs either in the spleen or lymph node data set when training the predictive model using input biomarkers from the 7-UP main panel and the 7-UP alternative panel, respectively. Note that the values for our method are lower for MSE and higher for PCC than that 7-UP panels. (b) PCA of average marker intensity within each cell was done for each tissue, and the fraction of explained variance is show for each principal component. Note the simpler variance strucure for the spleen datasets.

which indicates those markers are in relatively unique modes and harder to predict. For example, both models are able to predict the location of CD34 expression but do not fully recover its shape. Even though CD15 is the worst prediction made by the lymph node model, its expression pattern is still predicted accurately with only a few spots missed. CD15 is also the one that expresses very different patterns compared to other markers measured in L1-norm in spleen tissue. The minimal difference (measured in L1-norm) between CD15 to other markers is 0.722. However, considering the model performance only after three iterations, the L1-norm between the predicted and real images of CD15 on the spleen validation set is 0.443. Also, the differences between CD15 and three markers in the input set (CD5, DAPI, and SMActin), the L1-norm measured on the spleen validation set are 0.722, 1.13, and 1.035, respectively. Therefore, it demonstrates that the model will indeed learn the marker expression pattern instead of simply producing a pattern similar to its input. Note that the sizes of the final selected minimal sets vary a lot across different tissues. Expression in spleen is relatively easy to predict compared to lymph node. To quantitatively show this, we then performed principal component analysis (PCA) on expression matrices of 2 tissues' test data sets. Figure 2 (b) shows the ratio of variance explained by each PC, where it is obvious to see the lymph node curve is flatter than the spleen curve. This suggests lymph node cells are more variable to some extent. From an information theory standpoint, to recover the same amount of information of the lymph node data matrix, more PCs are needed which requires more markers to observe and the task is more difficult than for spleen. This also explains the better performance we achieved on the spleen dataset, where more biomarkers' expressions are correlated.

Double-panel experiment. As proof of concept, we carried out experiments to test the distributed version of our proposed method on small and large intestine CODEX image datasets, where each image contains 46 measured biomarker expressions. For each dataset, we first made a holdout test set, then for the remaining images, we partitioned them into two folds as training and validation sets (see detailed in Methods). In Figure 4, we monitored the validation performances of models of two panels. In general, the tendencies of learning curves are similar. However, due to the difficulty of the prediction tasks are not guaranteed the same on both panels, we observed a gap between the curves of two panels for large intestine model. Figure 4 (b) shows a PCA of marker intensity of small and large intestine cells. We see that the biomarker expressions of small intestine cells are the most difficult to predict as the first PC only explains less than 44% of variance and it also requires more observed markers for the prediction. Also, prediction tasks on both intestine datasets are harder compared to the spleen and lymph node datasets as the variance explained by PCs has a longer tail. That is, for small and large intestine datasets, the first 10 PCs explain 75.26% and 83.57% variance, respectively, while 96.32% and 90.71% variance is explained in spleen and lymph node datasets.

We trained the small and large intestine models for 18 iterations, yielding 18 biomarkers as the minimal set. For large intestine, the selected markers were: CD8, ITLN1,

DRAFT

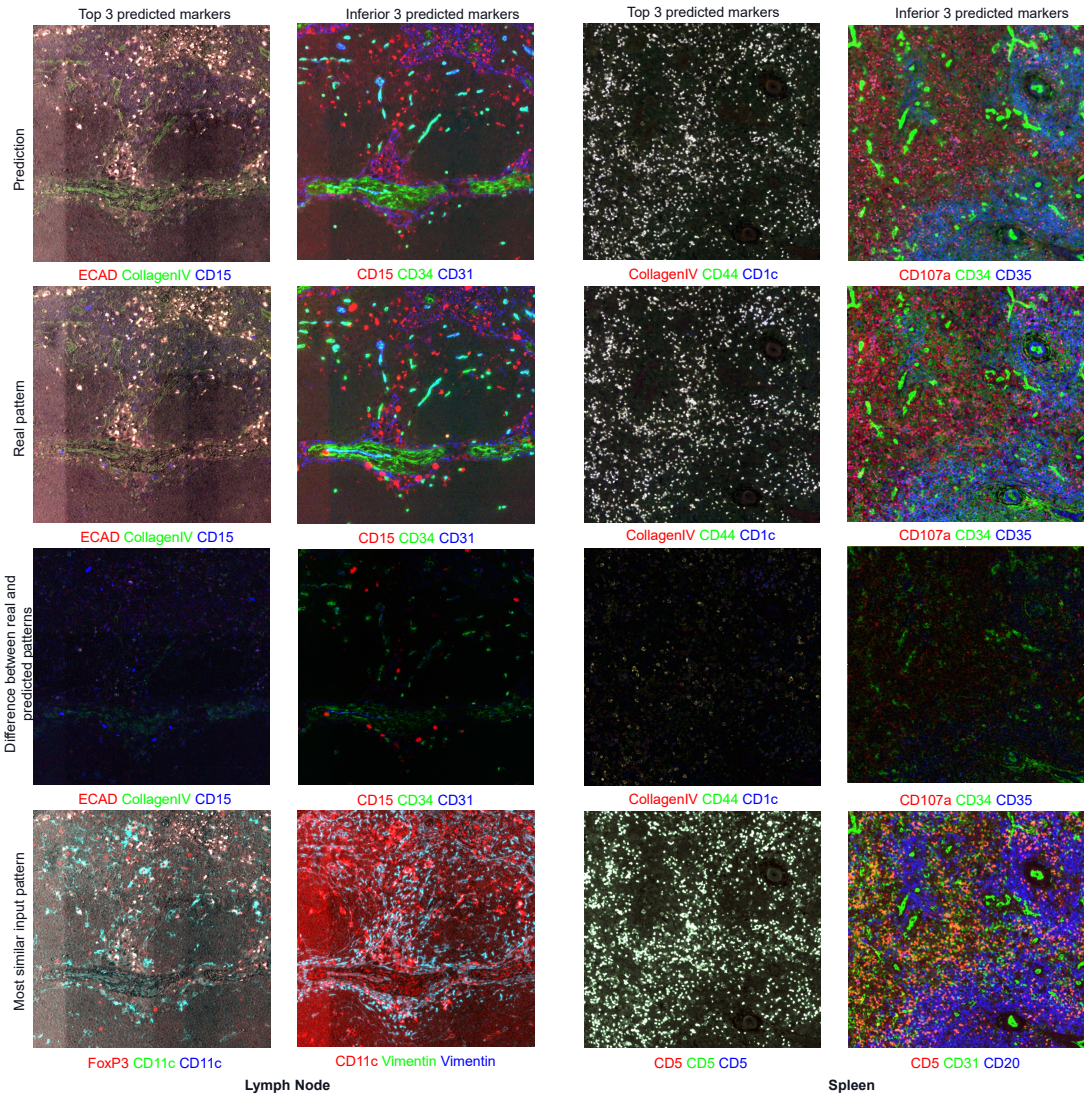

Fig. 3. Examples of patches from synthetic and real images for top and inferior 3 biomarkers for the lymph node and spleen test sets. The marker names below each panel show which markers are assigned to each RGB channel. The absolute difference between the predicted and real images are shown in the third row. The fourth row shows the expression patterns of markers in the selected minimal set that are most similar to the corresponding image patches shown in the second row. For example, for the lymph node, FoxP3 is the marker selected whose pattern is most similar to ECAD's. These illustrate whether accurate prediction can be made only by mimicking the input pattern or whether learning informative latent patterns from the input is required.

CD49a, DRAQ5, CD57, CD90, aSMA, Vimentin, HLADR, SOX9, CD4, CD44, CD21, CD15, MUC1, Hoechst, CDX2, and CD56. Among them, 11 markers were selected from panel A, 12 markers were selected from panel B, and there were 5 shared markers selected out of 6. Similar to Figure 3, examples of comparison of top and inferior predictions with their corresponding real images are shown in Figure 5. When examined visually, even with relatively limited number of biomarkers, the large intestine model captures most of tissue structures. For top predictions, as shown in the first column of Figure 5, the model can predict signals well, the differences between prediction and real image are only several small blobs in the signal of CD138 and CD3. Even considering the worst prediction examples shown in the second column, we observe that the model only does not predict the correct shape of the lower part structure labeled by MUC2, but predicts the locations well. And the model is able to make good predictions of the upper part structure labeled by MUC2. A possible reason is that the texture of the MUC2 signal varies at different regions of the tissue, as the real image pattern shown in the second column of Figure 5, the red signal is denser in the bottom part compared to the top part. However, when looking at the input ITLN1 expression pattern which is most similar to MUC2, the texture is consistent in two parts. Therefore, a convolution-based model with a limited perception field did not capture the variation of the MUC2 expression pattern at different locations. For small intestine, the selected markers were: CD117, CD8, Ki67, CD36, CD45RO, CD49a, DRAQ5, CD90, SOX9, CD44, Hoechst, aSMA, CD49f, Vimentin, cytokeratin, ITLN1, CD31, and MUC1. Among them, 12 markers were selected from panel A, 12 markers were selected from panel B, and all 6 shared markers were selected. Even though the small intestine is the hardest one to

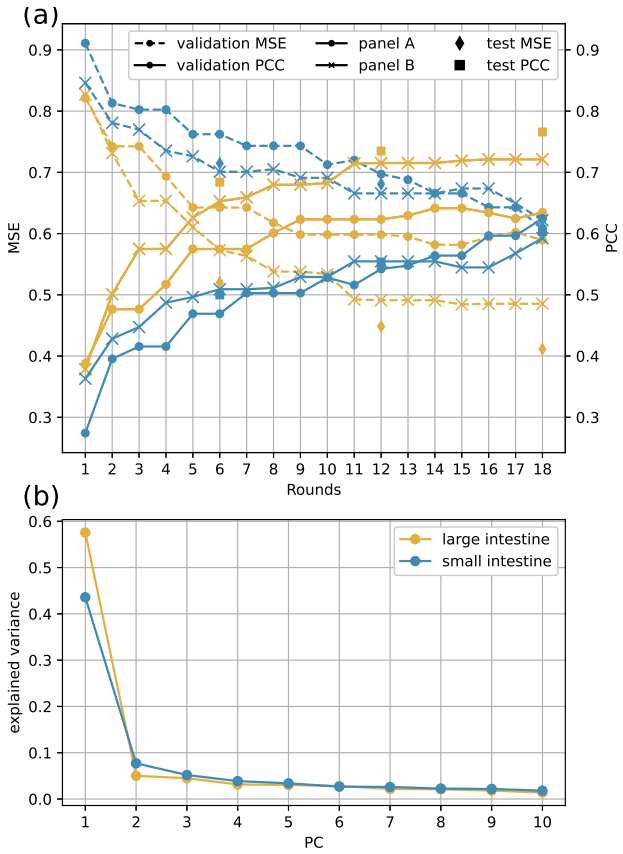

Fig. 4. (a) Similar to Figure 2, curves for MSE and PCCs between predicted and real images from validation sets are shown for the two panels. Model performances on the test sets are shown in diamonds and squares. (b) Variance explained by each PC for single-cell level expression of large and small intestine cells is shown for the test sets of the two tissues.

predict among the four datasets we present, the third column of Figure 5 shows that the model can still predict the location of most of the signals, but it is hard to recover the shape of cells. When looking at the corresponding input pattern, the correlation between the prediction target and input is weak, which requires the model to learn a complex relationship. Also, the signal of marker expression in the small intestine tissue typically distributes very sparsely, which poses difficulty to train a neural network by a pixel level objective. An extreme example is the marker MUC2, the worst predicted marker of the small intestine model; the highest Pearson correlation of its pattern to the input patterns is smaller than 0.1, as shown in the second and fourth rows of the second column of Figure 5. Although the model performs worst for this marker, the Pearson correlation between the predicted pattern to the real pattern is higher than 0.1.

Similar to the previous analyses on cell type identification on predicted biomarker expression, we show the results of large and small intestine models in Table 2. The cell segmentation and clustering methods used here were the same as in the previous part. It is worth noting that the small intestine model performs better than the large intestine model, suggesting that there is more redundancy in converting the pixel-level information to a 'higher' single-cell level for small intestine images compared to large intestine images. For example, since we use the average marker intensity in a given cell region as the marker expression for that cell, so it omits the locations and shapes of the expression pattern in the cell, but they are captured by pixel-level evaluations.

Discussion

We have developed novel algorithms for the significant problem of optimizing protein marker sets for multiplexed tissue imaging. Despite recent advances in multiplexed imaging techniques, the number of protein biomarkers can be imaged in the same tissue sample is still limited. This makes the selection of marker panels a critical step that defines how much information can be obtained. Our method is based upon the expectation that some marker intensities can be predicted from other measured marker intensities. Our results show that models trained on panels selected by our method can produce accurate predictions of images of unmeasured marker intensities, and also outperform a previous approach for the related problem of predicting cell-level intensities.

Selecting efficient marker sets is especially important since many thousands of proteins can be expressed in any given cell and current technologies do not allow that many to be imaged in a single sample. We have therefore demonstrated that our method can be extended to learning an informative subset even when the full set cannot be imaged in the same sample.

In the future, our approach can be potentially improve by using alternative deep learning networks, such as generative adversarial networks and self-attention based models that capture long-range interaction between pixels. Additional directions for optimizing marker panels include integrating other types of proteomics or genomics information into the marker graph.

Our results strongly suggest the feasibility of constructing spatial networks for all proteins without imaging them in the same sample.

DRAFT

Funding

This work was supported in part by a grant from the National Institutes of Health Common Fund, OT2 OD026682.

Bibliography

1. John W Hickey, Elizabeth K Neumann, Andrea J Radtke, Jeannie M Camarillo, Rebecca T Beuschel, Alexandre Albanese, Elizabeth McDonough, Julia Hatler, Anne E Wiblin, Jeremy Fisher, et al. Spatial mapping of protein composition and tissue organization: a primer for multiplexed antibody-based imaging. *Nature Methods*, 19(3):284–295, 2022.
2. Noah F Greenwald, Geneva Miller, Erick Moen, Alex Kong, Adam Kagel, Thomas Dougherty, Christine Camacho Fullaway, Brianna J McIntosh, Ke Xuan Leow, Morgan Sarah Schwartz, et al. Whole-cell segmentation of tissue images with human-level performance using large-scale data annotation and deep learning. *Nature biotechnology*, 40(4):555–565, 2022.
3. Tianyi Zhao and Zhaozheng Yin. Weakly supervised cell segmentation by point annotation. *IEEE Transactions on Medical Imaging*, 40(10):2736–2747, 2020.
4. Eric Czech, Bulent Arman Aksoy, Pinar Aksoy, and Jeff Hammerbacher. Cytokit: a single-cell analysis toolkit for high dimensional fluorescent microscopy imaging. *BMC bioinformatics*, 20(1):1–13, 2019.
5. Kevin J Cutler, Carsen Stringer, Teresa W Lo, Luca Rappez, Nicholas Stroustrup, S Brook Peterson, Paul A Wiggins, and Joseph D Mougous. Omnipose: a high-precision morphology-independent solution for bacterial cell segmentation. *Nature methods*, pages 1–11, 2022.
6. Zafran Hussain Shah, Marcel Müller, Tung-Cheng Wang, Philip Maurice Scheidig, Axel Schneider, Mark Schüttpelz, Thomas Huser, and Wolfram Schenck. Deep-learning based denoising and reconstruction of super-resolution structured illumination microscopy images. *Photonics Research*, 9(5):B168–B181, 2021.

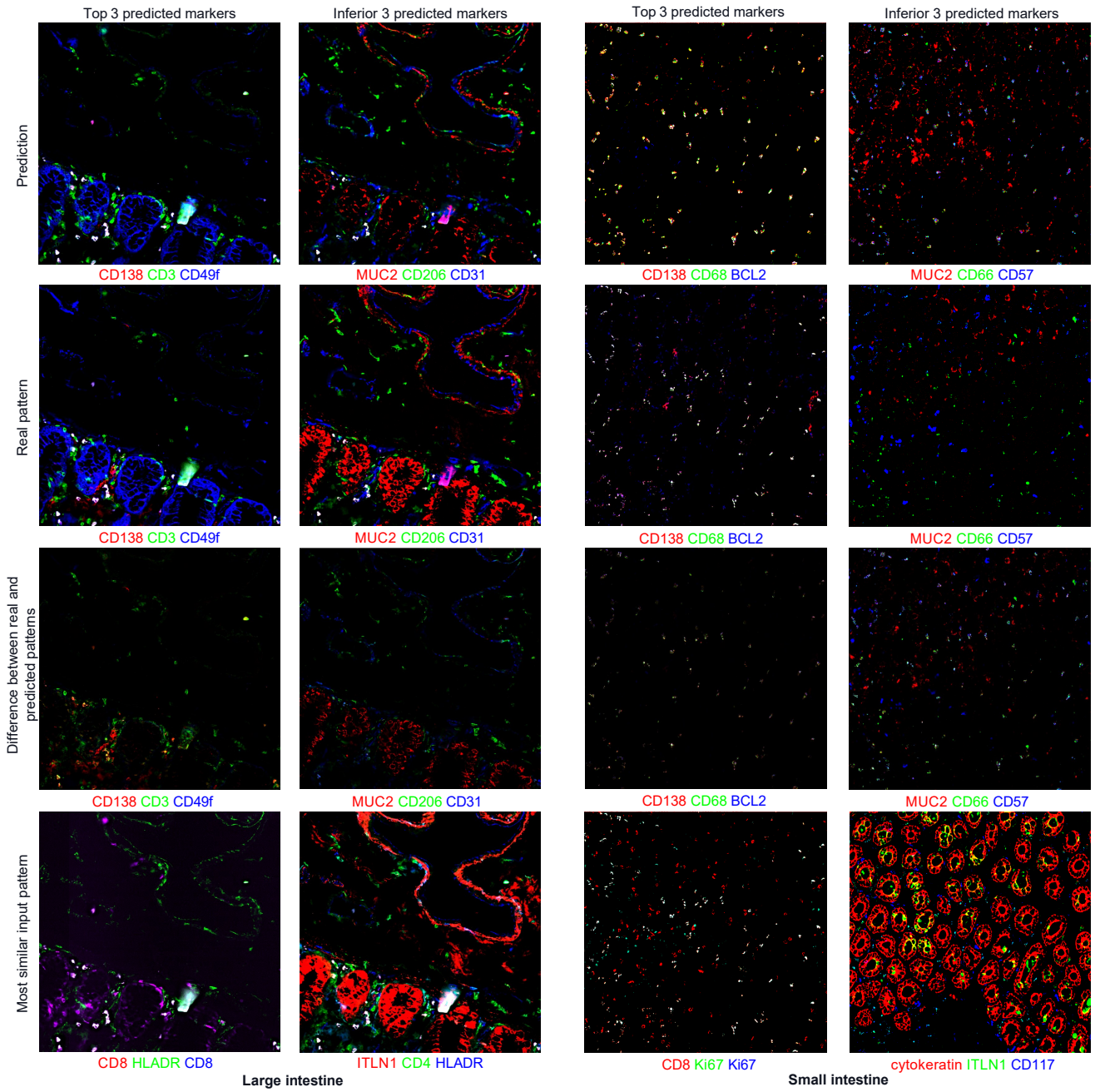

Fig. 5. Example images for the two panel study. Similar to Figure 3, example patches from predicted and real images for top and inferior 3 biomarkers are shown for the large and small intestine datasets. Also shown are the absolute values of the differences between the prediction and real images and the marker from the selected minimal set which is most similar to the markers shown in the second row.

Dataset	Large intestine						Small intestine					
Number of cell types	5		10		15		5		10		15	
	0.771	0.512	0.816	0.550	0.835	0.542	0.800	0.530	0.840	0.574	0.861	0.590

Table 2. Rand Index (left) and normalized mutual information (right) between cell type assignments using synthetic images compared to real images for large and small intestine datasets.

7. Chang Qiao, Di Li, Yong Liu, Siwei Zhang, Kan Liu, Chong Liu, Yuting Guo, Tao Jiang, Chuyu Fang, Nan Li, et al. Rationalized deep learning super-resolution microscopy for sustained live imaging of rapid subcellular processes. *Nature biotechnology*, pages 1–11, 2022.
8. Maria Brbić, Marinka Zitnik, Sheng Wang, Angela O Pisco, Russ B Altman, Spyros Darmanis, and Jure Leskovec. Mars: discovering novel cell types across heterogeneous single-cell experiments. *Nature methods*, 17(12):1200–1206, 2020.
9. Maria Brbić, Kaidi Cao, John W Hickey, Yuqi Tan, Michael P Snyder, Garry P Nolan, and Jure Leskovec. Annotation of spatially resolved single-cell data with stellar. *Nature Methods*, pages 1–8, 2022.
10. John W Hickey, Yuqi Tan, Garry P Nolan, and Yury Goltsev. Strategies for accurate cell type identification in codex multiplexed imaging data. *Frontiers in Immunology*, page 3317, 2021.
11. Chawin Ounkomol, Sharmishtaa Seshamani, Mary M Maleckar, Forrest Collman, and Gregory R Johnson. Label-free prediction of three-dimensional fluorescence images from transmitted-light microscopy. *Nature methods*, 15(11):917–920, 2018.
12. Eric M Christiansen, Samuel J Yang, D Michael Ando, Ashkan Javaherian, Gaia Skibinski, Scott Lipnick, Elliot Mount, Alison O'neil, Kevan Shah, Alicia K Lee, et al. In silico labeling: predicting fluorescent labels in unlabeled images. *Cell*, 173(3):792–803, 2018.
13. Zhengyang Wang, Yaochen Xie, and Shuiwang Ji. Global voxel transformer networks for

augmented microscopy. *Nature Machine Intelligence*, 3(2):161–171, 2021.
14. Huangqingbo Sun, Xuecong Fu, Serena Abraham, Shen Jin, and Robert F Murphy. Improving and evaluating deep learning models of cellular organization. *Bioinformatics*, 38 (23):5299–5306, 10 2022. ISSN 1367-4803. doi: 10.1093/bioinformatics/btac688.
15. Yair Rivenson, Hongda Wang, Zhensong Wei, Kevin de Haan, Yibo Zhang, Yichen Wu, Harun Günaydın, Jonathan E Zuckerman, Thomas Chong, Anthony E Sisk, et al. Virtual histological staining of unlabelled tissue-autofluorescence images via deep learning. *Nature biomedical engineering*, 3(6):466–477, 2019.
16. Dan Li, Hui Hui, Yingqian Zhang, Wei Tong, Feng Tian, Xin Yang, Jie Liu, Yundai Chen, and Jie Tian. Deep learning for virtual histological staining of bright-field microscopic images of unlabeled carotid artery tissue. *Molecular imaging and biology*, 22(5):1301–1309, 2020.
17. Yijie Zhang, Kevin de Haan, Yair Rivenson, Jingxi Li, Apostolos Delis, and Aydogan Ozcan. Digital synthesis of histological stains using micro-structured and multiplexed virtual staining of label-free tissue. *Light: Science & Applications*, 9(1):1–13, 2020.
18. Parmida Ghahremani, Yanyun Li, Arie Kaufman, Rami Vanguri, Noah Greenwald, Michael Angelo, Travis J Hollmann, and Saad Nadeem. Deep learning-inferred multiplex immunofluorescence for immunohistochemical image quantification. *Nature Machine Intelligence*, 4 (4):401–412, 2022.
19. Edwin Yuan, Magdalena Matusiak, Korsuk Sirinukunwattana, Sushama Varma, Łukasz Kidziński, and Robert West. Self-organizing maps for cellular in silico staining and cell substate classification. *Frontiers in Immunology*, page 4437, 2021.
20. Eric Wu, Alexandro E Trevino, Zhenqin Wu, Kyle Swanson, Honesty J Kim, H Blaize D'Angio, Ryan Preska, Gregory W Charville, Piero D Dalerba, Umamaheswar Duvvuri, et al. 7-up: generating in silico codex from a small set of immunofluorescence markers. *bioRxiv*, 2022.
21. Muhammed Fatih Balın, Abubakar Abid, and James Zou. Concrete autoencoders: Differentiable feature selection and reconstruction. In *International conference on machine learning*, pages 444–453. PMLR, 2019.
22. HuBMAP Consortium. The human body at cellular resolution: the nih human biomolecular atlas program. *Nature*, 574(7777):187–192, 2019.
23. Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
24. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
25. Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2(1): 193–218, 1985.
26. Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: is a correction for chance necessary? In *Proceedings of the 26th annual international conference on machine learning*, pages 1073–1080, 2009.

DRAFT