Orals

Jiangmei (Ruby) Xiong

2023-08-15

Table of contents

Statistical Analysis in Multiplexed Immunofluorescence Imaging

This is the document for doctoral oral exam of Jiangmei Xiong. This document will walk you through the basics of multiplexed immunofluorescence image, Jiangmei's first dissertation paper, and a literature review for missing data imputation in multiplexed immunofluorescence imaging.

1 Chapter 1: Introduction to Multiplexed Immunofluorescence Images

1.1 Multiplexed Immunofluorescence Images

Multiplexed Immunofluorescence (mIF) Image is a recent development from Immunofluorescence (IF), a branch of Immunohistochemistry (IHC). The first structural conceptualization of IHC is established in 1941. Coons, Creech, and Jones (1941) described that in formalin-prefixed mammalian tissue, there is a type of antibody that can be identified by fluorescent antigens. Since then, IHC is developed into an important tool for cancer diagnosis (Duraiyan et al. 2012). Within the word "immunohistochemistry", "immuno" refers to the antigen-antibody reaction in the process, "histo" means tissue, and "chemistry" is the process. During IHC, antibody can be tagged with labels such as enzyme, fluorochromes, which reacts when the corresponding antigen-antibody bind is formed (Ramos-Vara 2005). Similarly, the word immunofluorescence can split into "immuno" and "fluorescence", and "fluorescence" corresponds to the fluorescent signal generated by the fluorocromes(Hussaini, Seo, and Rich 2022).

IHC/IF can only detect one biomarker for a tissue region. This limitation makes IHC/IF unable to identify more complicated expression patterns that require more than one biomarker (Sheng et al. 2023). The development of multiplexed IHC (mIHC)/IF(mIF) image resolved this issue. Multiplexed IHC/IF image display different protein information for each plex of the image, while retaining the spatial and morphological information of the tissue (Eng et al. 2022). mIHC/mIF

can be seen as a stack of images, each presenting a different portion of the same tissue.

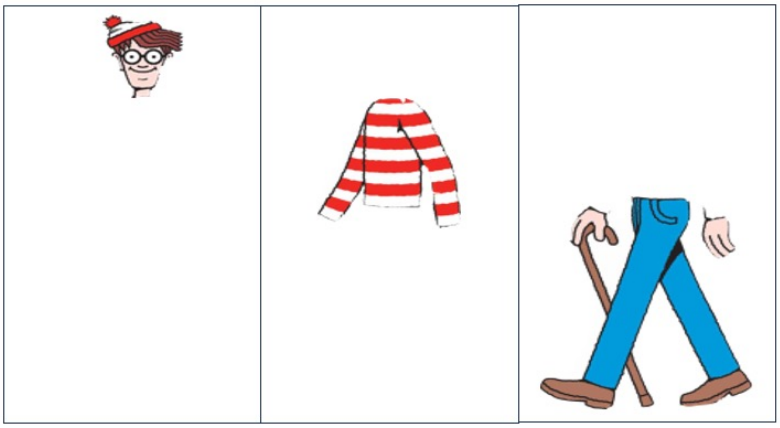

Figure 1.1: This is what it might look like if Waldo is to be analyzed in mIF style (??).

Figure 1.2 by Sheng et al. (2023) shows several different methods for creating mIF images. It can be seen that the first two methods both uses cycles of stain-photo-removal, and the last method is a one-off step where all labels are tagged at once. Sheng et al. (2023) also tabulated all multiplexd IHC/IF technologies, where the number of biomarkers that can be identified ranges from 4 to 100.

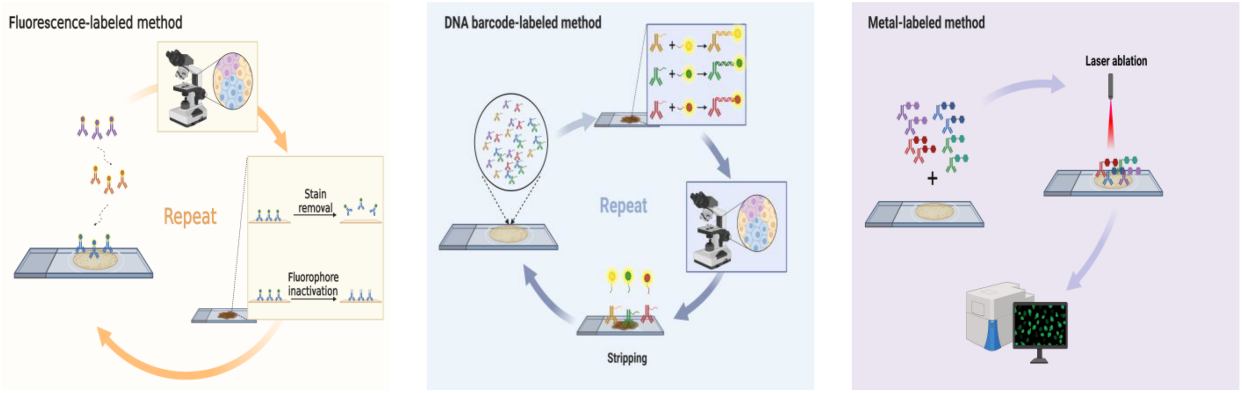

Figure 1.2: Different methods for creating mIF/mIHC images. Image courtesy of Sheng et al. (2023)

mIHC/mIF images are widely used in studies for immune tumor microenvironments (iTME). The studies often involves cell type proportions within a certain region, spatial clustering of immune cells or spatial interaction among different cell types (Wrobel, Harris, and Vandekar 2023). For example: Schürch et al. (2020) discovered that within granulcyte cell meightborhood, the enrichment PD-1+CD4+ T cells are correlated with the survival outcome of a subset of colorectal cancer patients; B. Chen et al. (2021) shows different immune cell proportion and clustering between different colorectal tumor types; Steinhart et al. (2021) found that certain immune cell proportions and spatial interactions are correlated with ovarian cancer patient survival outcomes.

1.2 Data Structure

For mIHC/mIF images, each individual "plex" corresponds to a different immune protein identified by a type of stain. Each plex goes through analogous transition as shown in Figure 1.3 and form a table of cell expressions, and the tables of different protein expressions are combined in the end.

Initially, greyscale intensity is assigned to each pixel. The greyscale intensity is taken as the intensity of marker expression. The image then goes through cell segmentation, which are usually based on machine learning or deep learning methods(McKinley et al. 2022; Schüffler et al. 2015). DAPI, a fluorescent stain typically used for cell morphology identification, is most often used for the initial cell segmentation. The same cell-segmentation will be used for all other marker channels. Next, the pixel intensities, pixel positions, and the cell that the pixel belongs to are entered into a dataframe. Finally, the pixel intensities and position will be averaged within the cell group. Often, the median of pixel intensities is used as well, to reduce the impact of pixel intensity outliers. The end result after combining all marker channels will be a dataset with each row representing an individual cell, columns of different marker expressions, and cell properties such as position, cell type (e.g. tumor cell or not, tissue type).

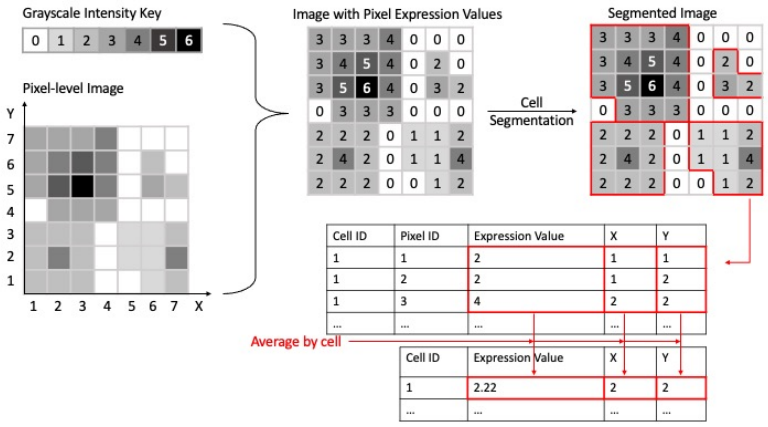

Figure 1.3: Transition from a single-plex mIHC/mIF image to a single-cell dataframe. The greyscale intensity range is only a demonstration. In application, the range of the greyscale intensity depends on the configuration of image softwares. The cell intensity of the bottom left cell is shown in the table as an example.

2 Chapter 2: GammaGateR

2.1 Research question

Before important spatial insights can be gleaned using statistical methods, mIF images undergo an intensive preprocessing pipeline to obtain single-cell measurements. While there are various steps included in the pipeline such as image registration, single-cell segmentation, quantification, and batch correction, cell phenotyping is typically the final step before downstream analyses on the cell-level data, similarly to other single-cell assays. Cell phenotyping identifies individual cell phenotypes from the measured marker expression values of the cell and directly affects the subsequent cell population analysis results.

The two most common approaches for cell phenotyping in mIF are manual gating and graph-based multivariate clustering. In manual gating, each sample is visualized separately to determine a threshold, and super-threshold cells are labeled as marker positive. This procedure is repeated for all marker channels and slides, and the phenotypes are determined by combining combinations of marker-positive cells. Alternatively, multivariate graph-based clustering is adapted from other single-cell assays. This approach first performs cell clustering, then assigns a phenotype to each cell group based on their average expression profile. Multivariate graph-based clustering is implemented with various modifications across many software packages. Unfortunately, both methods are labor intensive, and their accuracy suffers from image noise and spatial artifacts in mIF images that cause marker expression histograms to appear continuous or uni-modal . As a result, both phenotyping methods possess shortcomings that cannot be ignored. On one hand, manual gating can be subjective. On the other hand, graph-based clustering results are prone

to over-clustering and producing poor separation between clusters.

2.2 Previous works

The challenges described above are well recognized and there are a few methods and software developed that attempt to automate cell phenotyping for mIF images. For example, CellSighter is a recently proposed supervised deep-learning algorithm for cell phenotyping that requires a "gold standard" training dataset. Another recent solution, ASTIR (Automated assignment of cell identity from single-cell multiplexed imaging and proteomic data), is a fast unsupervised approach that defines cell phenotypes from segmented cell-level data by using a neural network-based mixture model assuming a multivariate log-normal distribution. Instead of binary outputs like in classification methods, ASTIR returns posterior probabilities of different cell types for each cell. This type of output is advantageous because it offers more information than nominal cell types and leaves cell labeling to the clinician's discretion. Lastly, Ahmadian et al. treat the analysis as a pixel classification problem and design a single-step framework for mIF phenotyping that is integrated with other preprocessing steps.

Nevertheless, inconsistencies persist in the results rendered by these learning-based methods when applied across markers, slides, batches, and datasets. These inconsistencies result from the immense variation in the cell-level distribution of phenotyping markers that are often too nuanced to be removed by existing batch correction methods. For these reasons, it is difficult to fully automate the cell phenotyping process, despite the availability of automated tools, and manual gating is still used to perform cell phenotyping because it is easy to visualize and evaluate the quality of the phenotype.

2.3 Methods

Because automated methods cannot be run without evaluation and supervised methods require a gold-standard dataset, no

method is truly fully automated. As a solution, we develop an explicitly semi-automated algorithm called GammaGateR. GammaGateR allows the user to easily perform cell phenotyping, visualize results, and conduct interpretable quality control while reducing manual labor. Based on a novel closed-form Gamma mixture model (cfGMM), GammaGateR is a probabilistic model that is fitted to each channel and slide separately, and outputs positive-component probabilities for each marker. These can then be easily thresholded and combined for semi-automated marker gating or input directly into downstream analysis. GammaGateR has important technical advantages, including 1) improved computation time and model convergence due to its novel closed-form property, and 2) high consistency and reproducibility for phenotyping results across mIF data batches due to incorporation of parameter boundary constraints. In applications on real-world mIF data, we find that GammaGateR has fast and consistent results across many slides and markers. We provide an open-source implementation of our method in the new GammaGateR R package (https://github.com/jiangmeirubyxiong/gammagater).

2.3.1 GammaGateR

The GammaGateR algorithm is unique to existing methods for its focus on parsimoniously modeling cell-level marker expression densities. This approach yields tailored-to-slide model estimation in cell-level mIF data where marker expression distributions can vary substantially across slides. The algorithm uses a zero-inflated two-component GMM to model marker expression for each slide. The Gamma mixture model naturally identifies marker-positive and marker-negative cell distributions and returns the probability of belonging to the marker-positive cell distribution for each cell. The returned probabilities can either be used directly in subsequent analysis or combined and dichotomized to define cell phenotypes. GammaGateR incorporates user-specified constraints to provide consistent model fit across a large number of slides. The model evaluation methods included in GammaGateR allow the user to evaluate the constraints and quality check results. The power source of GammaGateR is the closed-form Gamma mixture model, which is

a novel approach to phenotyping for mIF data that makes it more computationally efficient than traditional GMMs.

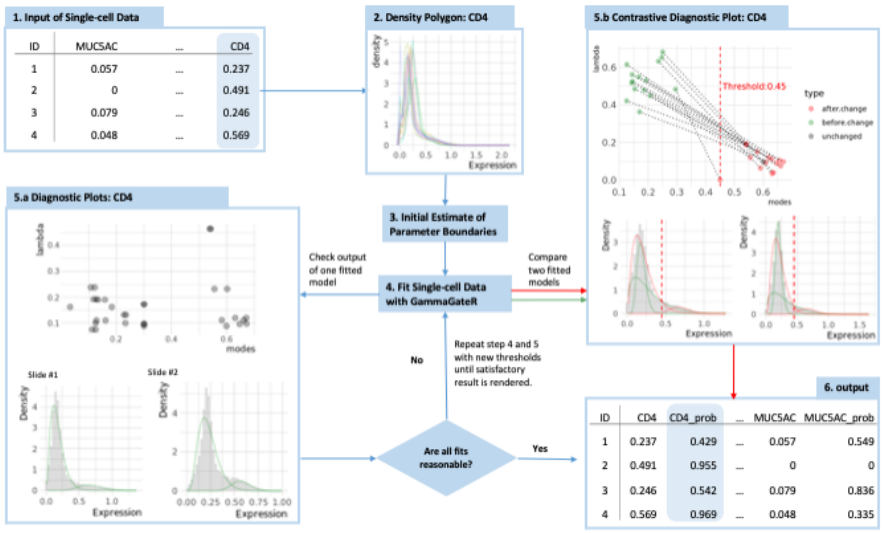

Figure 2.1: Figure 1

2.3.2 cfGMM

For mIF data, we use the GMM to fit cell marker expression values as a weighted sum of different probability distributions that represent unique cell populations [35]. The Gamma distribution is an excellent model for marker values because the domain of the Gamma distribution is strictly positive and it has the flexibility to model the varying skewed densities seen in mIF marker values (Figure 1.5.a). However, GMMs are not scalable for mIF image data, because they rely on computationally inefficient numerical methods to obtain the maximum likelihood estimator (MLE). The slow convergence of the MLE for the GMM makes it prohibitive to apply across a large number of channels, slides, and cells. As a solution, we develop a closed-form GMM (cfGMM; https://github.com/jiangmeirubyxiong/cfgmm) estimation procedure based on a recently developed estimator for the Gamma distribution [36]. In addition, to improve computational efficiency, the cfGMM has the benefit of allowing prior constraints on model parameters. With the cfGMM in

GammaGateR, we enable the flexibility to include a biologically meaningful range for the mode of each component in the Gamma mixture model. This way, users of GammaGateR can restrict estimation to biologically meaningful values.

2.3.2.1 Derivation

We assume the data is a random sample $x_1, \ldots, x_n$ from a K component generalized gamma mixture distribution. The density function of X is

$$P(X = x) = \sum_{k=1}^{K} \lambda_k f(x; a_k, b_k, \gamma_k).$$

and the log-likelihood of the dataset is

$$\ell(\mathbf{x}|\mathbf{a}, \mathbf{b}, \boldsymbol{\lambda}) = \sum_{i=1}^{n} \log \left\{ \sum_{k=1}^{K} \lambda_k f(x_i|a_k, b_k) \right\} \tag{2.1}$$

For each generalized gamma component k, $\lambda_k \in [0, 1]$ are the mixture parameters, $\sum_k \lambda_k = 1$; f denotes the generalized gamma density function; a_k, b_k, γ_k are the parameters for the generalized gamma.

Here, we use the expectation maximization (EM) algorithm [dempster_maximum_1977] for parameter estimation. EM algorithm is a standard approach for parameter estimation in mixture models. It introduces the latent multinomial variable $Z_i = (Z_{i1}, \ldots, Z_{iK})$ into the model and maximizes the expected value of the complete data likelihood [dempster_maximum_1977]. The expectation of the complete data likelihood to be maximized for the generalized gamma distribution is

$$\mathbb{E}_Z \ell(x \mid Z) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \log f(x_i; a_k, b_k, \gamma_k),$$

where

$$z_{ik} = \mathbb{P}(Z_{ik} = 1 \mid x_i; \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{\gamma}) = \frac{f(x_i|a_k, b_k, \gamma_k)}{\sum_{j=1}^{K} f(x_i|a_j, b_j, \gamma_k)}, \tag{2.2}$$

$\mathbf{a} = (a_1, a_2, ..., a_K)$, and $\mathbf{b}$, $\boldsymbol{\lambda}$ are similarly defined vectors.

From here, the maximization of the expectation is now analogous to the maximization of generalized gamma distribution for each component of the mixture model.

The expectation of log-likelihood is

$$\mathbb{E}_{z|x}[\log(L(\mathbf{x}|\mathbf{z}))] = \sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik} \log f_k(x_i) \tag{2.3}$$

and

$$f_k(x) = G(a_k, b_k, \gamma_k) = \frac{\lambda_k x^{a_k\gamma_k - 1} exp\{(-x/b_k)^{\gamma_k}\}}{b_k^{a_k\gamma_k}\Gamma(a_k)} \tag{2.4}$$

where $\gamma_k = 1$.

By eqref{eq1}, eqref{eq2} The expected joint log-likelihood is

$$\mathbb{E}_{z|x}[\log(L(\mathbf{x}|\mathbf{z}))] = \sum_{k=1}^{K}\sum_{i=1}^{n} z_{ik}\left(\log\gamma_k - a_k\gamma_k \log b_k - log\Gamma(a_k) + (a_k\gamma_k - 1)\log X_i - (\frac{X_i}{b_k})^{\gamma_k}\right) \tag{2.5}$$

The estimators of each of the K terms of the expected joint log-likelihood are derived as follows:

first take derivative of the expression from eqref{eq3}

$$\sum_{i=1}^{n} z_{ik}\left(\log\gamma_k - a_k\gamma_k\log b_k - \log\Gamma(a_k) + (a_k\gamma_k - 1)\log X_i - \left(\frac{X_i}{b_k}\right)^{\gamma_k}\right)$$

with respect to a_k, b_k, γ_k separately:

$$\frac{\partial\mathbb{E}_{z|x}[\log(L(\mathbf{x}|\mathbf{z}))]}{\partial a_k} = \sum_{i=1}^{n} z_{ik}(-\psi(a_k) - \gamma_k\log b_k + \gamma_k X_i) = 0 \tag{2.6}$$

Note that $\psi(x) = \dfrac{d}{dx}\log\Gamma(x)$ is digamma function.

$$\frac{\partial \mathbb{E}_{z|x}[\log(L(\mathbf{x}|\mathbf{z}))]}{\partial b_k} = \sum_{i=1}^{n}(z_{ik})(-a_k\gamma_k/b_k + \gamma_k X_i^{\gamma_k} b_k^{-\gamma_k-1}) = 0 \tag{2.7}$$

$$\frac{\partial \mathbb{E}_{z|x}[\log(L(\mathbf{x}|\mathbf{z}))]}{\partial \gamma_k} = \sum_{i=1}^{n} z_{ik}\left(\frac{1}{\gamma_k} - a_k\log b_k + a_k \log X_i - \left(\frac{X_i}{b_k}\right)^{\gamma_k}\log\frac{X_i}{b_k}\right) = 0 \tag{2.8}$$

Among which, eqref{eq5} can be solved as

$$\hat{b}_k(a_k,\gamma_k) = \left(\frac{\sum_{i=1}^{n} z_{ik}X_i^{\gamma_k}}{a_k\sum_{i=1}^{n} z_{ik}}\right)^{1/\gamma_k} \tag{2.9}$$

Substitute eqref{eq7} into eqref{eq6}:

$$
\begin{aligned}
&\frac{\partial \mathbb{E}_{z|x}[\log(L(\mathbf{x}|\mathbf{z}))]}{\partial \gamma_k} \\
&= \sum_{i=1}^{n} z_{ik}/\gamma_k + \sum_{i=1}^{n} a_k z_{ik} \log(\frac{X_i}{b_k}) - \sum_{i=1}^{n} z_{ik} (\frac{X_i}{b_k})^{\gamma_k} \log(\frac{X_i}{b_k}) \\
&= \sum_{i=1}^{n} z_{ik}/\gamma_k + \sum_{i=1}^{n} a_k z_{ik} (\log X_i - \log b_k) - b_k^{-\gamma_k} \sum_{i=1}^{n} z_{ik} X_i^{\gamma_k} (\log X_i - \log b_k) \\
&= \sum_{i=1}^{n} z_{ik}/\gamma_k + \sum_{i=1}^{n} a_k z_{ik} \log X_i - \log b_k \sum_{i=1}^{n} a_k z_{ik} - b_k^{-\gamma_k} \sum_{i=1}^{n} z_{ik} X_i^{\gamma_k} \log X_i + b_k^{-\gamma_k} \log b_k \sum_{i=1}^{n} z_{ik} X_i^{\gamma_k} \\
&= \sum_{i=1}^{n} z_{ik}/\gamma_k + \sum_{i=1}^{n} a_k z_{ik} \log X_i - \log b_k \sum_{i=1}^{n} a_k z_{ik} - b_k^{-\gamma_k} \sum_{i=1}^{n} z_{ik} X_i^{\gamma_k} \log X_i \frac{a_k \sum_{i=1}^{n} z_{ik}}{\sum_{i=1}^{n} z_{ik} X_i^{\gamma_k}} \log b_k \sum_{i=1}^{n} z_{ik} X_i^{\gamma_k} \\
&= \sum_{i=1}^{n} z_{ik}/\gamma_k + \sum_{i=1}^{n} a_k z_{ik} \log X_i - \log b_k \sum_{i=1}^{n} a_k z_{ik} - b_k^{-\gamma_k} \sum_{i=1}^{n} z_{ik} X_i^{\gamma_k} \log X_i + a_k \sum_{i=1}^{n} z_{ik} \log b_k \\
&= \sum_{i=1}^{n} z_{ik}/\gamma_k + a_k \sum_{i=1}^{n} z_{ik} \log X_i - \frac{a_k \sum_{i=1}^{n} z_{ik}}{\sum_{i=1}^{n} z_{ik} X_i^{\gamma_k}} \sum_{i=1}^{n} z_{ik} X_i^{\gamma_k} \log X_i \\
&= \sum_{i=1}^{n} z_{ik}/\gamma_k + a_k \left(\sum_{i=1}^{n} z_{ik} \log X_i - \frac{\sum_{i=1}^{n} z_{ik}}{\sum_{i=1}^{n} z_{ik} X_i^{\gamma_k}} \sum_{i=1}^{n} z_{ik} X_i^{\gamma_k} \log X_i \right) = 0
\end{aligned}
$$

Solving this, we have

$$
\hat{a}_k(\gamma_k) = \frac{\sum_{i=1}^{n} \frac{z_{ik}}{\gamma_k}}{\frac{\sum_{i=1}^{n} z_{ik}}{\sum_{i=1}^{n} z_{ik} X_i^{\gamma_k}} \sum_{i=1}^{n} z_{ik} X_i^{\gamma_k} \log X_i - \sum_{i=1}^{n} z_{ik} \log X_i} \quad (2.10)
$$

Plug $\gamma_k = 1$ in eqref{eq8}, we now have

$$\begin{aligned}
\hat{a}_k(\gamma_k = 1) &= \frac{\sum_{i=1}^{n} z_{ik}}{\frac{\sum_{i=1}^{n} z_{ik}}{\sum_{i=1}^{n} z_{ik} X_i} \sum_{i=1}^{n} z_{ik} X_i \log X_i - \sum_{i=1}^{n} z_{ik} \log X_i} \\
&= \left(\frac{\sum_{i=1}^{n} z_{ik} X_i \log X_i}{\sum_{i=1}^{n} z_{ik} X_i} - \frac{\sum_{i=1}^{n} z_{ik} \log X_i}{\sum_{i=1}^{n} z_{ik}} \right)^{-1} \\
&= \frac{\sum_{i=1}^{n} z_{ik} \sum_{i=1}^{n} z_{ik} X_i}{\sum_{i=1}^{n} z_{ik} \sum_{i=1}^{n} z_{ik} X_i \log X_i - \sum_{i=1}^{n} z_{ik} \log X_i \sum_{i=1}^{n} z_{ik} X_i}
\end{aligned} \tag{2.11}$$

$$\begin{aligned}
\hat{b}_k(\hat{a}_k, \gamma_k = 1) &= \frac{\sum_{i=1}^{n} z_{ik} X_i}{\hat{a}_k \sum_{i=1}^{n} z_{ik}} \\
&= \frac{\sum_{i=1}^{n} z_{ik} \sum_{i=1}^{n} z_{ik} X_i \log X_i - \sum_{i=1}^{n} z_{ik} \log X_i \sum_{i=1}^{n} z_{ik} X_i}{\left(\sum_{i=1}^{n} z_{ik} \right)^2}
\end{aligned} \tag{2.12}$$

In addition, $\hat{\lambda}_k$ can simply be estimated as

$$\hat{\lambda}_k = \frac{\sum_{i=1}^{n} z_{ik}}{n} \tag{2.13}$$

It is worth noting that we are not maximizing the exact Gamma distribution, therefore the algorithm we devise here is an EM-type algorithm.

2.4 Simulation

To compare the bias and compute time of the closed-form GMM to maximum likelihood GMM implementation, we run the cfGMM, the constrained cfGMM, and the GMM to evaluate bias and variance in a sample size of 10,000 across 1,000 simulations. We simulate a two-component mixture model with parameters $\boldsymbol{\lambda} = (0.3, 0.7), \boldsymbol{a} = (0.5, 8), \boldsymbol{b} = (0.5, 1/3)$. For the constrained estimator, we restrict the mode of each component to be in the range $(-\infty, 0)$ and $(0, 5)$ for marker negative and marker positive components, respectively, which include the true mode for each component, 0 (no mode) and 7/3.

Both closed-form estimation procedures have substantially faster computation time than the MLE (Figure 2a) while maintaining similarly low bias (Figure 2b). The sample size used in the simulation is roughly similar to that of the cell-level mIF image dataset, which further proves that cfGMM brings computation efficiency to our target application. The closed-form GMM, therefore, enables computationally feasible, precise, and flexible model estimation when applied to a large number of channels and slides using GammaGateR. It is also worth noting that the constrained cfGMM converges slightly faster than without constraints. This implies that when using cfGMM, computational cost can be reduced with proper knowledge of biological priors.

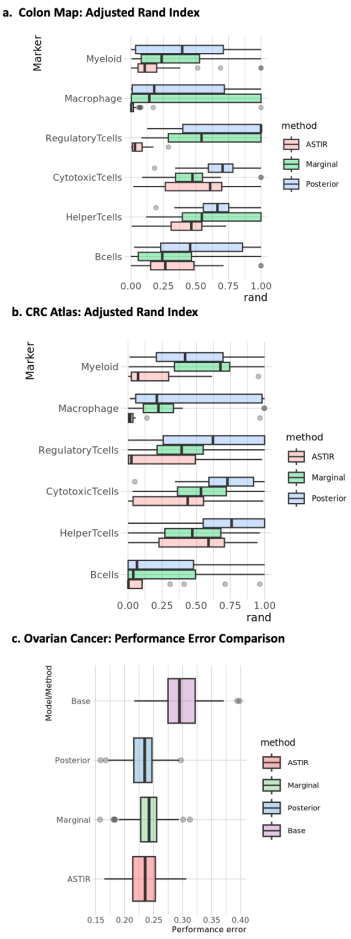

Figure 2.2: Figure 2

2.5 Analysis

3 Chapter 3: Missing data imputation in mIF imaging

Just like all data created and collected by human being, missing data is inevitable in mIF image as well. Bao et al. (2021) in their paper gave a brief summary of types of missing data in mIF image, as in Figure 3.1. Case 1 in Figure 3.1 refers to the missing of one or more entire marker channel. This type of missing data occurs but rarely, often due to low image quality. Other possible reasons for missing channel, not described in Bao et al. (2021), can be supply shortage in certain type of fluorescent material or change in research plan. Despite the rarity of case 1, there are demand for marker channel imputation with low-plex images. Due to time and financial constraints, mIF with no more than seven channels are often more feasible to obtain than the 40-channel mIFs(Wu et al. 2023). To break the restraint in obtaining cell phenotypes from few number of markers, imputation of marker channels are proposed. Case 2 Figure 3.1 occurs more frequently, when tissue wears off in the cycles of staining - wash off described in Figure 1.2.

Owning to the rapid development in the field of computer vision, all current applications in mIF imputation are implemented with machine learning and/or deep learning methods. In the three applications covered in this document today, Bao et al. (2021) uses generative adversarial networks (GANs), Wu et al. (2023) uses gradient boosting decision tree in combination with convolutional neural network, and Sims and Chang (2023) uses masked autoencoders (MAE). All methods preforms ideally well, as expected out of the maturity of machine learning methods. However, the subsequent analysis can benefit from statistical thinking in data imputation. This will be discussed further in chapter 4.

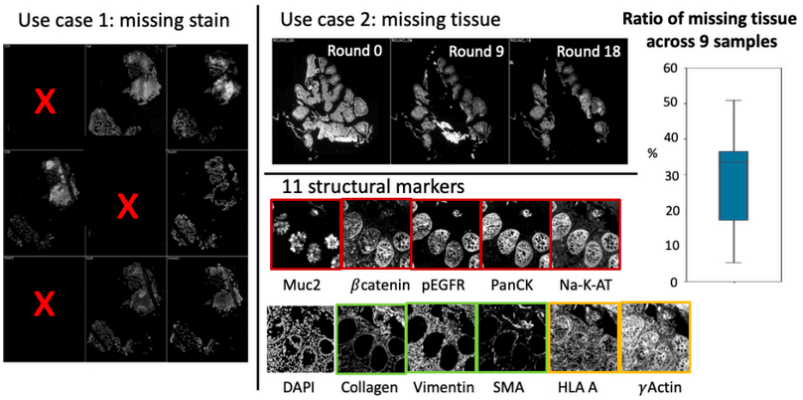

Figure 3.1: Types of missing data in mIF. Image courtesy of Bao et al. (2021).

3.1 Application case 1: Missing tissue imputation

3.1.1 Method: GANs

The fundamental version of GANs comprises of two compartments: a discriminator and a generator (Goodfellow et al. 2014). Figure 3.2 by Bok and Langr (2019) gives a brief sketch of how GANs works. Like a turn-based strategy game, the two components take turns to run an epoch. Starting with a noise distribution (usually a uniform distribution), the generator's goal is to generate data that is close to the real data. The discriminator's goal is to identify the real data between a mix of real data and data generated by the discriminator. With classification error feed back to generator and discriminator, both opponents update their weights: generator will try to maximize the probability that discriminator misclassify generated data as real, and the discriminator will try to maximize classification accuracy. Within infinite number of rounds, they will eventually reach a state close to equilibrium, where either party can only improve negligibly: generator generates close-to-real data, and discriminator classifies with 50% accuracy(Bok and Langr 2019). This is the point where the algorithm stops.

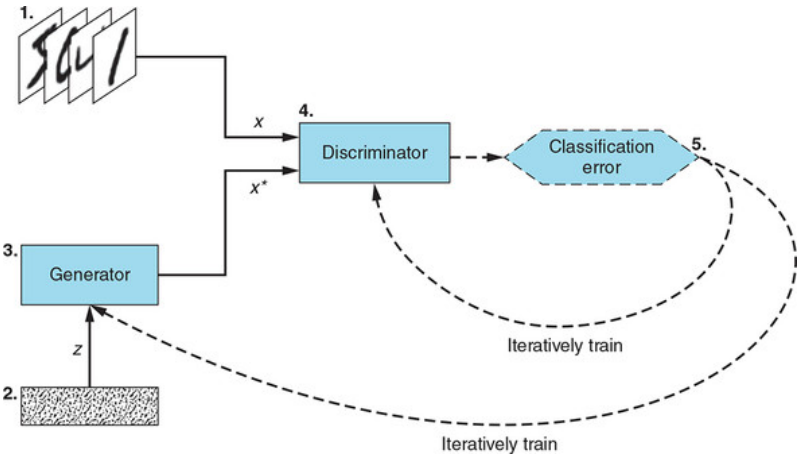

Figure 3.2: How GANs work. Image courtesy of Bok and Langr (2019)

One disadvantage of the original GANs is its weak control on the generated data, due to the random noise input. This disadvantage stands out especially with image synthesis. Conditional GANs (CGANs) provided a promising solution to this issue by including a condition $\boldsymbol{y}$ on both generator and discriminator (Mirza and Osindero 2014). $\boldsymbol{y}$ is usually data from the same class, for example other images in the case of image synthesis. Based on this, *pix2pix* is able to perform image-to-image translation by using image pairs to train the data, where one image in the pair serves as input while the other image serves as the output(Isola et al. 2017; Souza et al. 2023).

3.1.2 Application in mIF: pixN2N-HD

pixN2N-HD is a "novel multi-channel high-resolution image synthesis approach". "N2N" represents "N-to-N", which distinguishes itself from the widely used (N-1)-to-1 model. N represents the number of marker channels, and in the dataset used in this paper, N=11. In (N-1)-to-1 design, 10 channels are used as input and 1 channel is used as output, and this repeats for 11 permutations of models. The "N-to-N" instead uses a random gate strategy, as shown in Figure 3.3. This strategy randomly selects up to N-1=10 markers as the "missing" data. Blank im-

ages are input to the generator, where it generates image for all channels, but only imputed image for missing channel is sent to the discriminator. The discriminator will attempt to discriminate the real and fake image, similar as described above. the image input for generator also serves as the condition for the discriminator, similar to *pix2pix*.

This paper evaluated the model performance by comparing "N-to-N" model with "(N-1)-to-1" model and another "(N-1)-to-1 random gate" model, which blends in random gate but still needs to train 11 separate models. An index for measuring image similarity, the structure similarity index measure (SSIM) is used to assess whether "N-to-N" model generates comparable results with the other two methods (Wang et al. 2004). The result shows that all pairs of methods do not have significantly different results on a 0.05 significance level, and therefore the methods are concluded to be comparable. This "N-to-N" model take significantly less amount of time to train compared to the other methods, which is very meaningful in terms of effective computation.

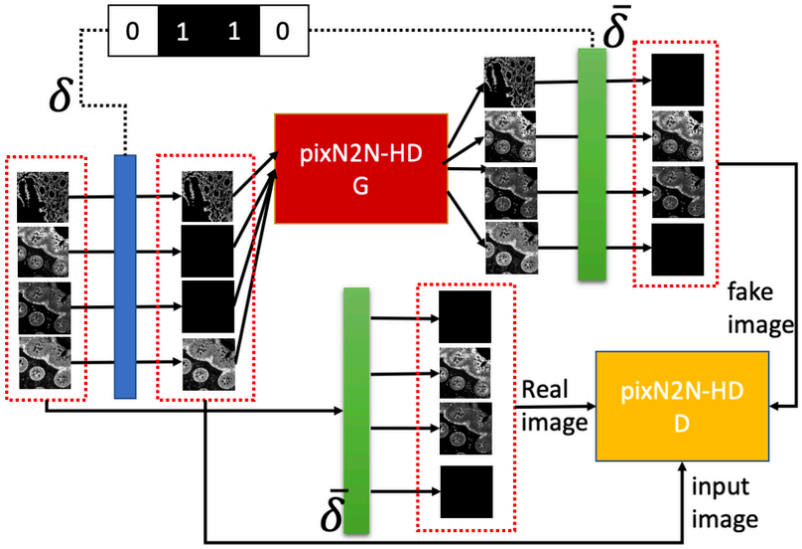

Figure 3.3: Work flow of pixN2N-HD. Image courtesy of Bao et al. (2021).

3.2 Application case 2: Marker channel imputation

Both 7-UP and CyCIF panel reduction are intended for marker channel imputation, providing access to otherwise expensive high-plex (40+ channels) mIF image for study that can only obtain low-plex images. Interestingly, the two application uses very different methods for imputation.

3.2.1 Application 2.1: 7-UP

7-UP starts from a 7-plex mIF image and generates high-plex image that can identify up to 16 different cell types (Wu et al. 2023). This approach consists of three main parts:

1. Marker panel selection. This part will select the seven markers to start with, using concrete autoencoder. Concrete autoencoder is an feature selection method, of which the loss function is the difference between the original sample and the reconstructed low-dimension sample (Balın, Abid, and Zou 2019).
2. Morphology feature extraction. This step uses a convolutional neural network to learn the morphology features, i.e. spatial and structural features of cells. Convolutional neural networks are similar to layers of linear regressions, where there are more combinations of weights linked to each input variable.
3. Marker expression imputation. Once the location and structure of cells are learned, the important task left is to impute the expression of each marker on each cell. The imputation is performed using XGBoost, a scalable gradient-boosting tree software (T. Chen and Guestrin 2016).

A series of evaluation and analysis are performed to show the validity of the method. The performance of the method is examined in three ways:

1. Calculating the pearson correlation coefficient between the imputed marker expression and the testing data marker expression.
2. Calculating the F1 score between the imputed and testing data cell type. F1-score is the harmonic mean of precision and sensitivity: $2/(sensitivity^{-1}+precision^{-1})$. Cell type is generated from the marker expression through k-nearest neighbor.
3. Patient survival status, HPV status and disease recurrence are used to further evaluate the cell type outcomes. AUC score for patient status prediction is calculated for both imputed data outcome and training data.

All evluation shows that the imputation generates comparable results with the training data, hence proven the validity of this method.

3.2.2 Application 2.2: CyCIF panel reduction

This method is intended to be an improvement from their own previous work (Ternes et al. 2022). The previous work first go through panel selection and then imputes marker channel with variatioal autoencoder. The current improved method (Sims and Chang 2023) uses masked autoencoder for image synthesis as shown is Figure 3.4. The difference is the adoption of within-model iterative selection of marker panels, as the authors believe that panel selection should be more closely tied with panel reconstruction. Starting with standard DAPI, each marker is added to the panel, predict marker intensities of other panels, and mean Spearman correlation is calculated between the predicted intensity and real intensity. The marker with highest correlation is selected, and the next round continues until the panel is constructed. The ratio of masked channels depends on tasks, though 25%~75% is a reasonable range.

The method outcome is evaluated by Spearman correlation with the true data. It is shown in the results that both MAE and the iterative panel selection outperforms the VAE and out-of-box panel selection of the previous method.

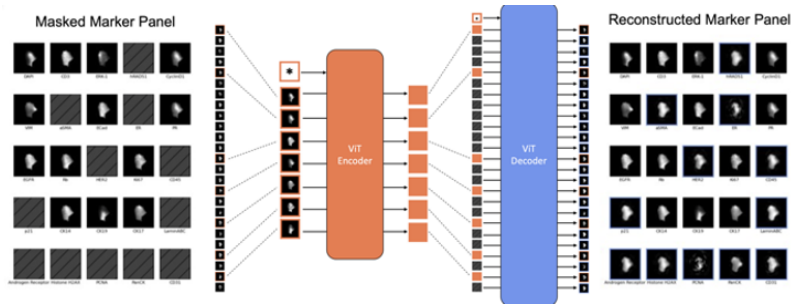

Figure 3.4: CyCIF panel reduction with autoencoder. Figure courtesy of Sims and Chang (2023).

4 Chapter 4: Future directions

The missing data in mIF image can be seen as missing completely at random (MCAR). In the missing data scenarios described in Chapter 3, none of them can be predicted. However, for the low-plex image channel imputation methods, the markers are selected by machine learning methods to maximize the accuracy. For this type of imputation, the imputation results should be taken with a grain of salt as this is missing not at random (MNAR). It might be possible to deduce the amount of bias this creates through statistical inference, but for now this is not in the scope of this project.

All methods in Chapter 3 evaluated the accuracy of their result by either comparing with the imputation outcome of a previous method or with a held-out evaluation dataset. In addition, Wu et al. (2023) in Section 3.2.1 used the imputation output for cell phenotyping and predicting patient phenotypic outcomes. In principle, this is not the best practice of subsequent analysis with the imputation outcome. Such prediction should be performed with multiple imputation at least, to obtain a reasonable estimation of confidence intervals.

Rubin (1996) states that it is important to be statistically valid for estimates of scientific estimands, such as population mean. Statistical validity for an estimand means an at least approximately non-biased point estimate, and an statistical test that rejects the null hypothesis no less than 5% of the time, when the nominal significance level is 5% (Rubin 1996; Van Buuren 2018). Under such guideline, we can check the multiple imputation estimand's expectation and variance. Van Buuren (2018) presented a clear interpretation of such, conditioning on observed data: Let Y_{mis} be missing data, and Y_{obs} be observed data, Q be the population value and $\hat{Q}$ is the estimate of Q.

The posterior distribution of Q given observed data is

$$(Q|Y_{obs}) = \int P(Q|Y_{obs}, Y_{mis})P(Y_{mis}|Y_{obs})dY_{mis}$$

The posterior mean of Q is therefore

$$E[Q|Y_{obs}] = E[E(Q|Y_{obs}, Y_{mis})|Y_{obs}]$$

Which can be interpreted as the average of estimates over repeatedly imputed data, given the observed data.

The posterior variance is therefore

$$Var(Q|Y_{obs}) = E[Var(Q|Y_{obs}, Y_{mis})|Y_{obs}]+Var[E(Q|Y_{obs}, Y_{mis})|Y_{obs}]$$

Which can be interpreted as the within-variance of the estimate in each imputed data, and the between-variance among the repeatedly imputed data.

In this case, single imputation would be an unbiased estimator. However, it will be biased in variance estimation, as it does not incorporate between-variance at all. Therefore, for better accuracy with estimate variance, multiple imputation should be used. The subsequent project could use the colon map data in GammaGateR project, where some channels are missing, and compare the validity of confidence interval for the estimated quantities.

References

Balın, Muhammed Fatih, Abubakar Abid, and James Zou. 2019. "Concrete Autoencoders: Differentiable Feature Selection and Reconstruction." In *International Conference on Machine Learning*, 444–53. PMLR.

Bao, Shunxing, Yucheng Tang, Ho Hin Lee, Riqiang Gao, Sophie Chiron, Ilwoo Lyu, Lori A Coburn, et al. 2021. "Random Multi-Channel Image Synthesis for Multiplexed Immunofluorescence Imaging." In *MICCAI Workshop on Computational Pathology*, 36–46. PMLR.

Bok, Vladimir, and Jakub Langr. 2019. *GANs in Action: Deep Learning with Generative Adversarial Networks*. Simon; Schuster.

Chen, Bob, Scurrah Cherie'R, Eliot T McKinley, Alan J Simmons, Marisol A Ramirez-Solano, Xiangzhu Zhu, Nicholas O Markham, et al. 2021. "Differential Pre-Malignant Programs and Microenvironment Chart Distinct Paths to Malignancy in Human Colorectal Polyps." *Cell* 184 (26): 6262–80.

Chen, Tianqi, and Carlos Guestrin. 2016. "Xgboost: A Scalable Tree Boosting System." In *Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining*, 785–94.

Coons, Albert H, Hugh J Creech, and R Norman Jones. 1941. "Immunological Properties of an Antibody Containing a Fluorescent Group." *Proceedings of the Society for Experimental Biology and Medicine* 47 (2): 200–202.

Duraiyan, Jeyapradha, Rajeshwar Govindarajan, Karunakaran Kaliyappan, and Murugesan Palanisamy. 2012. "Applications of Immunohistochemistry." *Journal of Pharmacy & Bioallied Sciences* 4 (Suppl 2): S307.

Eng, Jennifer, Elmar Bucher, Zhi Hu, Ting Zheng, Summer L Gibbs, Koei Chin, and Joe W Gray. 2022. "A Framework for Multiplex Imaging Optimization and Reproducible Anal-

ysis." *Communications Biology* 5 (1): 438.

Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. "Generative Adversarial Nets." *Advances in Neural Information Processing Systems* 27.

Hussaini, Haizal Mohd, Benedict Seo, and Alison M Rich. 2022. "Immunohistochemistry and Immunofluorescence." In *Oral Biology: Molecular Techniques and Applications*, 439–50. Springer.

Isola, Phillip, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. "Image-to-Image Translation with Conditional Adversarial Networks." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1125–34.

McKinley, Eliot T, Justin Shao, Samuel T Ellis, Cody N Heiser, Joseph T Roland, Mary C Macedonia, Paige N Vega, Susie Shin, Robert J Coffey, and Ken S Lau. 2022. "MIRIAM: A Machine and Deep Learning Single-Cell Segmentation and Quantification Pipeline for Multi-Dimensional Tissue Images." *Cytometry Part A* 101 (6): 521–28.

Mirza, Mehdi, and Simon Osindero. 2014. "Conditional Generative Adversarial Nets." *arXiv Preprint arXiv:1411.1784*.

Ramos-Vara, Jose A. 2005. "Technical Aspects of Immunohistochemistry." *Veterinary Pathology* 42 (4): 405–26.

Rubin, Donald B. 1996. "Multiple Imputation After 18+ Years." *Journal of the American Statistical Association* 91 (434): 473–89.

Schüffler, Peter J, Denis Schapiro, Charlotte Giesen, Hao AO Wang, Bernd Bodenmiller, and Joachim M Buhmann. 2015. "Automatic Single Cell Segmentation on Highly Multiplexed Tissue Images." *Cytometry Part A* 87 (10): 936–42.

Schürch, Christian M, Salil S Bhate, Graham L Barlow, Darci J Phillips, Luca Noti, Inti Zlobec, Pauline Chu, et al. 2020. "Coordinated Cellular Neighborhoods Orchestrate Antitumoral Immunity at the Colorectal Cancer Invasive Front." *Cell* 182 (5): 1341–59.

Sheng, Wenjie, Chaoyu Zhang, TM Mohiuddin, Marwah Al-Rawe, Felix Zeppernick, Franco H Falcone, Ivo Meinhold-Heerlein, and Ahmad Fawzi Hussain. 2023. "Multiplex Immunofluorescence: A Powerful Tool in Cancer Immunotherapy." *International Journal of Molecular Sciences* 24 (4): 3086.

Sims, Zachary, and Young Hwan Chang. 2023. "A Masked Image Modeling Approach to Cyclic Immunofluorescence (CyCIF) Panel Reduction and Marker Imputation." *bioRxiv*, 2023–05.

Souza, Vinicius Luis Trevisan de, Bruno Augusto Dorta Marques, Harlen Costa Batagelo, and João Paulo Gois. 2023. "A Review on Generative Adversarial Networks for Image Generation." *Computers & Graphics*.

Steinhart, Benjamin, Kimberly R Jordan, Jaidev Bapat, Miriam D Post, Lindsay W Brubaker, Benjamin G Bitler, and Julia Wrobel. 2021. "The Spatial Context of Tumor-Infiltrating Immune Cells Associates with Improved Ovarian Cancer Survival." *Molecular Cancer Research* 19 (12): 1973–79.

Ternes, Luke, Jia-Ren Lin, Yu-An Chen, Joe W Gray, and Young Hwan Chang. 2022. "Computational Multiplex Panel Reduction to Maximize Information Retention in Breast Cancer Tissue Microarrays." *PLoS Computational Biology* 18 (9): e1010505.

Van Buuren, Stef. 2018. *Flexible Imputation of Missing Data*. CRC press.

Wang, Zhou, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. 2004. "Image Quality Assessment: From Error Visibility to Structural Similarity." *IEEE Transactions on Image Processing* 13 (4): 600–612. https://doi.org/10.1109/TIP.2003.819861.

Wrobel, Julia, Coleman Harris, and Simon Vandekar. 2023. "Statistical Analysis of Multiplex Immunofluorescence and Immunohistochemistry Imaging Data." In *Statistical Genomics*, 141–68. Springer.

Wu, Eric, Alexandro E Trevino, Zhenqin Wu, Kyle Swanson, Honesty J Kim, H Blaize D'Angio, Ryan Preska, et al. 2023. "7-UP: Generating in Silico CODEX from a Small Set of Immunofluorescence Markers." *PNAS Nexus* 2 (6): pgad171.