Comp683: Computational Biology

Lecture 16

March 25, 2025

Today

- Finish spatial phenotyping with LEAPH
- Start graph neural networks

CyTOF + Spatial Resolution

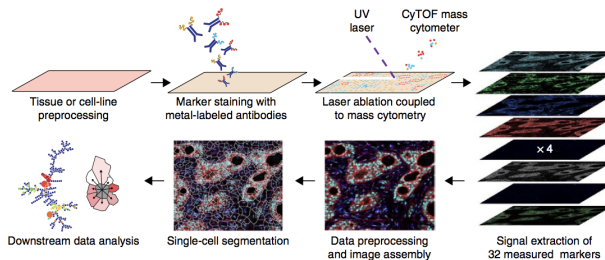An upgrade of regular CyTOF to image 32 proteins and their modifications at cellular resolution.

Figure: from Giesen *et al.* Nature Methods. 2016

Ultimately, an objective is to identify 'micro-environments' or spatially-localized subsets of cells with characteristic frequency patterns that are predictive of some outcome of interest.
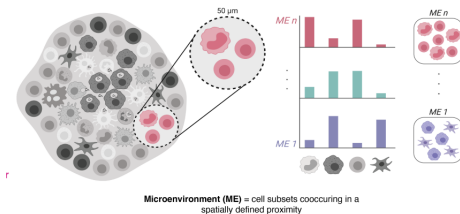

Microenvironment (ME) = cell subsets cooccuring in a spatially defined proximity

Figure: from https:
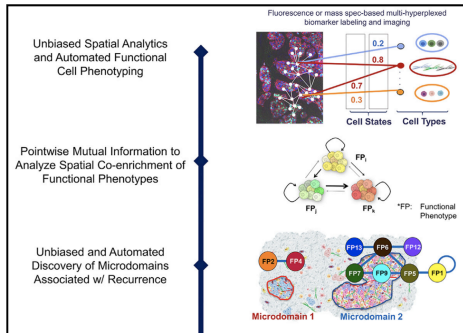//www.biorxiv.org/content/10.1101/2020.06.08.140426v1.full.pdf

Figure: from Furman *et al.* Cell Reports Methods. 2021. Functional responses of cells are determined both by internal state and interactions with neighbors and stimulus from local environment.

Notation in LEAPH

- For cell i, let its protein expression be represented as $\mathbf{x}_i \in \mathbb{R}^p$.
- Mixture of factors setup, with k dimensions in the latent space, with $\mathbf{x}_i = \Lambda \mathbf{z} + \boldsymbol{\mu} + \mathbf{v}$
  - Loadings in $\Lambda \in \mathbb{R}^{p \times k}$
  - Latent variables, $\mathbf{z} \in \mathbb{R}^{k \times 1}$. Assume zs generated from a standard normal with 0 mean and unit variance.
  - Noise term via, $\mathbf{v} \sim \mathcal{N}(0, \Psi)$
  - Mean vector, $\boldsymbol{\mu} \in \mathbb{R}^{p \times 1}$
  - Under this formulation, each $\mathbf{x}_i \sim \mathcal{N}(\mathbf{0}, \Lambda \Lambda^T + \Psi)$

Probabilistic Clustering Setup

The mixture of factor analyzers setup with the notation that we defined can be written as,

$$(\{\pi_j, \boldsymbol{\mu}_j, \Lambda_j\}_{j=1}^M\}, \Psi)$$

.

Note that π_j is the component or mixing weight for component j.

Mixture Model

Each $p(\mathbf{x}_i)$ is computed as

$$p(\mathbf{x}_i) = \sum_{j=1}^{M} \pi_j \mathcal{N}(\mathbf{x}_i \mid \boldsymbol{\mu}_j, \Lambda_j \Lambda_j^T + \Psi)$$

- π_j is the mixing weight for cluster j.

The parameters, λ and Ψ are updated in a close-form way using the EM algorithm. See here,
https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=766f4465747394d304d162197e091f1ae8f7f577.

Practicalities

- Overall, parameters being estimated are $\{\pi_j, \boldsymbol{\mu}_j, \Lambda_j\}_{j=1}^M, \Psi)$.
- They 2-dimensions for each latent space, so, $k = 2$.
- Ultimately, they get a prediction that each cell belongs to of the M components, and in particular for class j, $p(j \mid \mathbf{x}_i) = \frac{p(\mathbf{x}_i|j)p(j)}{\sum_{c=1}^M p(\mathbf{x}_i|c)p(c)}$
- Use the estimated probability between a cell i and a cluster c and create a matrix, $\Omega \in \mathbb{R}^{N \times M}$ where Ω_{ic} gives the probability that cell i belongs to cluster c.
- This gives a soft clustering interpretation for each cell.

Spatial Regularization Intuition

- Based on prior biological knowledge, there are known properties that for example, epithelial/tumor cells should be surrounded by or spatially proximal to other epithelial/tumor cells.

- There should also be some allowance for tumor-infiltrating cells, such as lymphocytes and other stromal cells.

A new Ω is optimized that encodes spatial information as follows,

$$\min_{\Omega} - \sum_{i=1}^{N} \sum_{j=1}^{M} \Omega_{ij} \log_2 \left(\Omega_{ij} \right) + \lambda \sum_{(m,n)} w_{mn} ||\Omega_m - \Omega_n||_2$$

Unpacking

$$\min_{\Omega} -\sum_{i=1}^{N}\sum_{j=1}^{M} \Omega_{ij} \log_2\left(\Omega_{ij}\right) + \lambda \sum_{(m,n)} w_{mn}||\Omega_m - \Omega_n||_2$$

- w_{jk} is a weight, calculate as the reciprocal of distance between cells j and k in the image
- The first term is basically an entropy term of ownership confidence
- The second term is promoting spatial coherence.
- λ controls the tradeoff between spatial coherence and membership confidence.

Effect of Spatial Regularization

In particular in the first example, a cell with a highly predicted assignment towards CP1 transitioned towards a phenotype of CP2 after spatial regularization.
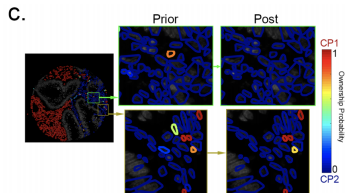

Figure: from Fig. 2 of https: //www.biorxiv.org/content/10.1101/2020.10.02.322529v3.full.pdf

Determining Specialized Cells

- Based on the Ω, assign each cell to one of the M phenotypes based on the j that gives the maximum probability.
- For a particular patient, p, create a feature vector $\mathbf{f}_p$ which gives the proportion of its cells assigned to each of the cell phenotypes.
- At times, the authors refer to specialized cell-types (membership probability $> 95\%$) in contrast to transitional and rare cells.

Recap and Transition

- The clustering part is straight-forward : Assume each cell is from one of M 2-dimensional latent factors
- Calculate a probability that each cell was from each of these latent factors
- Add penalties that enforce spatial coherence and certainty of assignment
- **Next step:** Identify microdomains with a collection of cells that are predictive of some penotype of interest.

Predicting Time to Recurrence in Colorectal Cancer

- Consider cohorts of patients with the following properties.
  - 45 patients in 'NED-8' category that have no evidence of disease for over 8 years
  - 46 patients in 'NED-3', where cancer came back within 3 years.

The goal is to translate the distributions of cell phenotypes that spatially co-occur to a signal that can be used for prediction.

Constructing a Cell Network For Each Patient

- Connectivity is determined by proximity in the image of the tissue
- For a pair of cells, m, and n, connect them with a weights, $w_{mn} = 1$ if their spatial distance, $d_{mn} < \tau$.
- Otherwise, $w_{mn} = 0$ and there are no edge between the cells

Identifying Spatial Co-Occurrence Between Cell Phenotype Pairs

Consider two phenotypes, f_i and f_j for a given set (e.g. a subset of patients, etc). The pairwise mutual information between these two phenotypes is defined as,

$$\text{PMI}_s\left(f_i, f_j\right) = \log_2\left(\frac{p\left(f_i^s, f_j^s\right)}{p\left(f_i^t\right) p\left(f_j^t\right)}\right)$$

- $p(f_i^s)$ is the probability of a particular phenotype, i occurring in a network set, s.
- $p(f_i^t)$ is the background probability of phenotype i.

Calculating Joint Phenotypic Probability for a Single Patient

Letting Ψ encode the set of edges for a particular patient, the joint probability of phenotypes i and j is given as,

$$p\left(f_i^s, f_j^s\right) = \frac{1}{z} \left(\sum_{(m,n) \in \Psi} w_{mn} \left(\vec{\Omega}_{mf_i} \vec{\Omega}_{nf_j} + \vec{\Omega}_{mf_j} \vec{\Omega}_{nf_i} \right) \right)$$

*Here z is a normalization over all combinations of i and j according to the computational phenotypes.

Specifying a Background Distribution

The background probability for a phenotype, i is simply the mean assignment probability over all cells, or,

$$p(f_i^t) = \frac{1}{N} \sum_{c=1}^{N} \Omega_{ci}$$

Ultimately, for each cell phenotype pair, (f_i, f_j) compute the PMI for each sample and consider how this relates to the patient re-occurence outcomes.

Looking at Significant Microdomains Between Groups

There were a few cellular phenotypes that tended to co-occur between the two patient groups.
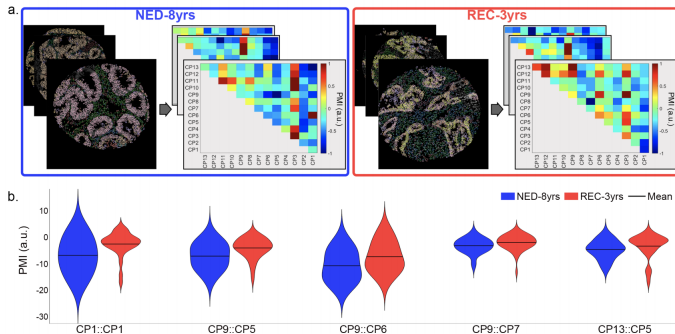

Figure: from Fig. 4 in https:
//www.biorxiv.org/content/10.1101/2020.10.02.322529v3.full.pdf

Switching gears to graph neural networks (GNNs) as another way to incorporate spatial information

GNN vs Simple Things

- You all love GNNs
- Recently there has been some work to study why GNNs are outperforming simpler methods and how to incorporate this intuition back to simpler methods
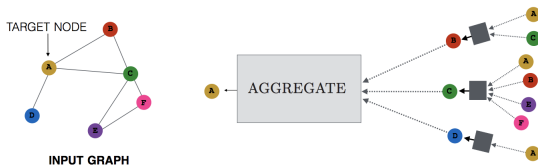

Figure: from https://www.cs.mcgill.ca/~wlh/grl_book/files/GRL_Book-Chapter_5-GNNs.pdf. Messages are aggregated from the neighborhood of some target node.

Neural Message Passing

- **Input:** $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a set of node features, $\mathbf{X} \in \mathbb{R}^{d \times |\mathcal{V}|}$
- **Output:** Node embeddings, $\mathbf{z}_u = \mathbf{h}_u^{(K)}, \forall u \in \mathcal{V}$, after K iterations
- **Each Message-Passing Iteration:** A hidden embedding $\mathbf{h}_u^{(k)}$ is updated according to information aggregated from node u's neighborhood, $\mathcal{N}(u)$.

Update Rule

$$\mathbf{h}_u^{(k+1)} = \text{UPDATE}^{(k)} \left(\mathbf{h}_u^{(k)}, \text{AGGREGATE}^{(k)} \left(\left\{ \mathbf{h}_v^{(k)}, \forall v \in \mathcal{N}(u) \right\} \right) \right)$$
$$= \text{UPDATE}^{(k)} \left(\mathbf{h}_u^{(k)}, \mathbf{m}_{\mathcal{N}(u)}^{(k)} \right)$$

- $\mathbf{m}_{\mathcal{N}(u)}^{(k)}$ is the *message* that is aggregated from u's graph neighborhood, $\mathcal{N}(u)$

- At each iteration, the AGGREGATE function takes as input the set of embeddings of the nodes in u's graph neighborhood, $\mathcal{N}(u)$ and applies it to a previous embedding $\mathbf{h}_u^{(k-1)}$ to generate the updated embedding $\mathbf{h}_u^{(k)}$)

[From Leskovec 224W 2021 slides]

Figure: from https:
//www.cs.cornell.edu/~arb/slides/2021-03-12-northeastern.pdf
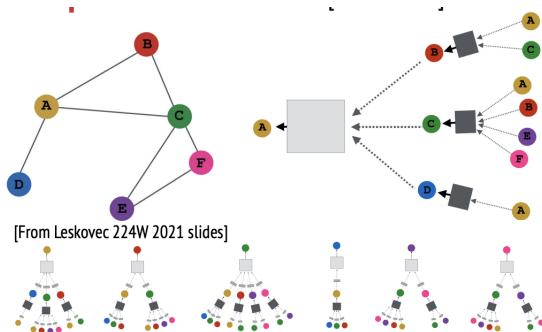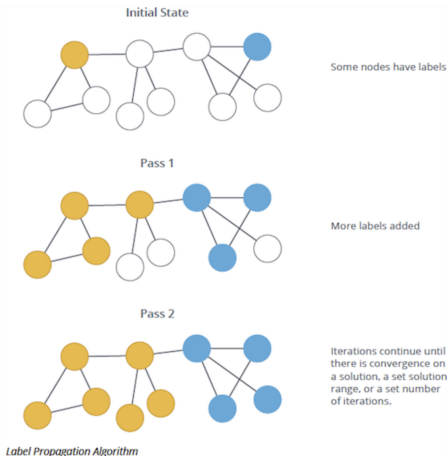
What is Label Propagation?

Some of your nodes are labeled, others are unlabeled, and you predict labels of unlabeled nodes based on the structure of the graph.

Label Propagation Algorithm

Label Propagation Formulation

Consider l labeled and u unlabeled nodes, where each node belongs to one of C classes. First define an $(l + u) \times (l + u)$ probabilistic transition matrix, T as,

$$T_{ij} = P(j \to i) = \frac{w_{ij}}{\sum_{k=1}^{l+u} w_{kj}}$$

.

Here, W is encoding our weighted adjacency matrix.

Also define a $(l + u) \times C$ label matrix, Y, where the ith row gives the probability toward each of the C clusters.

- $Y \leftarrow TY$. Update until convergence
- Row-normalize Y.
- Clamp the labeled data, or put all of the probability mass on the correct cluster index.

The Debate

1. Label Propagation [early 2000s]

[From Gleich & Mahoney 2015]
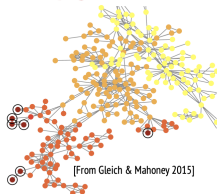
- Strong modeling assumption:
  connected nodes have similar labels.
- Works because of homophily [McPherson+ 01]
  a.k.a. assortativity [Newman 02]
- Why not use additional info/features?
- **FAST**
  a few sparse matrix-vector products

2. Graph Neural Networks [late 2010s]

[From Leskovec 224W 2021 slides]
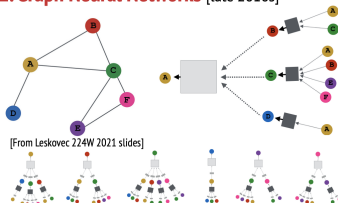
- Strong modeling assumption:
  labels only depend on neighbor features
- Works because these features are
  sometimes very informative.
- Why not assume labels are correlated?
- **SLOW**
  many parameters, irregular computation

8

Figure: from `https://www.cs.cornell.edu/~arb/slides/2021-03-12-northeastern.pdf`.
Tumor labels cannot be inferred based on cell-type frequencies, but instead should glean insights from the overall tissue architecture.
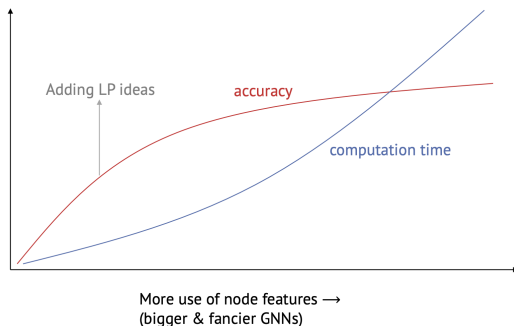
Tradeoffs

Adding LP ideas

accuracy

computation time

More use of node features →
(bigger & fancier GNNs)

Figure: from https:
//www.cs.cornell.edu/~arb/slides/2021-03-12-northeastern.pdf

Correct and Smooth Approach

- The goal is to compare how a couple of simple methods/intuition can be strung together can be used to classify nodes
- The main idea is to start with a cheap base prediction based on node features (e.g. attributes or coordinates of a spectral embedding), and clean up graph structure through label propagation (**correct and smooth**).

Three Step Process

1. A base prediction made with node features that ignores the graph structure (e.g. with a linear model)
2. A correction step which propagated uncertainties from the training data across the graph to correct the base prediction
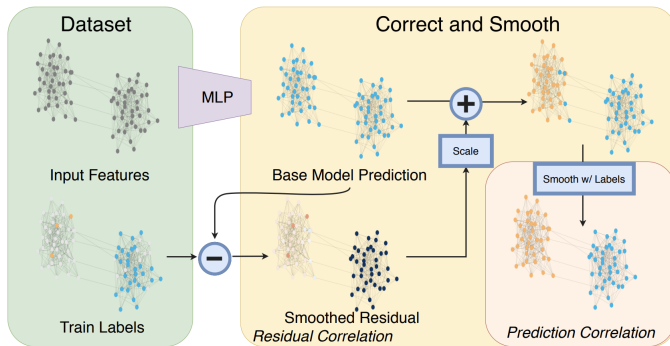3. A smoothing of the predictions over the graph.

Overview of Correct and Smooth Approach

Figure: from Huang *et al.* ICLR. 2021