UROP Notes

MICHAEL DIAO

1 RCTD

Our ultimate goal is to determine the fractional contributions of each cell type to a particular sample. We do this by maximum likelihood, using the following hierarchical model:

$$\begin{aligned} Y_{i,j} \mid \lambda_{i,j} &\sim \text{Poisson}(N_i \lambda_{i,j}) \\ \log \lambda_{i,j} &= \log(\vec{\beta}_i \cdot \vec{\mu}_j) + \alpha_i + \gamma_j + \varepsilon_{i,j}, \end{aligned}$$

where

- $Y_{i,j}$ is the random variable corresponding to the observed expression of gene j at pixel i,
- N_i is the number of transcripts for pixel i,
- $\vec{\beta}_i$ is the K-dimensional row vector of contributions from each cell type (where K is the number of cell types in question) at pixel i,
- $\vec{\mu}_j$ is the K-dimensional column vector of mean expressions of gene j for each cell type,
- α_i is a fixed pixel-specific effect.
- γ_j and $\varepsilon_{i,j}$ are random effects that introduce noise. γ_j in particular is intended to account for platform effects that may over- or underrepresent certain genes. We let these be normally distributed with mean 0 and variance σ_γ, σ_ε respectively.

Therefore, determining the fractional contributions of each cell type reduces to finding the maximum likelihood parameter $\vec{\beta}_i$ for each i.

> **Question 1.1.** Now we have a ton of parameters, potentially thousands. β alone introduces $K \times J$ of them. How do we do any useful estimation here?

We proceed in the following steps:

1. **Supervised estimation of cell type profiles**.

   Using a reference dataset, we estimate the parameters μ_j of expression levels for gene j, giving $\hat{\vec{\mu}}_j$ which will be used in the next steps.

   We can do this by obtaining a (e.g. scRNA-seq) reference annotated with cell types, after which $\vec{\mu}_j$ can be estimated as the empirical average normalized expression of gene j within each cell type.

2. **Gene filtering**.

   Using the estimated expression profiles $\hat{\vec{\mu}}_j$, we filter out genes that are not highly variable across cell types.

   We can do this by taking the expression profiles $\hat{\vec{\mu}}_j$ and selecting genes with a minimum average expression and sufficiently high variance.

3. **Platform Effect Normalization**.

   With an estimate for $\vec{\mu}_j$, it turns out we now have a way to estimate the platform effects γ_j as well. The idea is that we can consider the average observed expression across pixels

   $$\mathsf{M}_j = \frac{1}{I}\sum_{i=1}^{I} \mathsf{Y}_{i,j},$$

   whence

$$\begin{aligned}
\log \mathbb{E}_{\mathsf{M}_j|\vec{\lambda}_j}\left[M_j \mid \lambda_{1,j},\ldots,\lambda_{I,j}\right] &= \log\left(\frac{1}{I}\sum_{i=1}^{I} N_i\lambda_{i,j}\right) \\
&= \log\left(\frac{1}{I}\sum_{i=1}^{I} N_i \exp\Big(\log(\vec{\beta}_i\cdot\vec{\mu}_j)+\alpha_i+\gamma_j+\varepsilon_{i,j}\Big)\right) \\
&= \gamma_j + \log\left(\frac{1}{I}\sum_{i=1}^{I} N_i(\vec{\beta}_i\cdot\vec{\mu}_j)\exp\big(\alpha_i+\varepsilon_{i,j}\big)\right) \\
&= \gamma_j + \log\left(\frac{1}{I}\sum_{i=1}^{I}\left(\sum_{k=1}^{K}\beta_{i,k}\cdot\mu_{k,j}\right) N_i\exp\big(\alpha_i+\varepsilon_{i,j}\big)\right) \\
&= \gamma_j + \log\left(\sum_{k=1}^{K}\mu_{k,j}\sum_{i=1}^{I}\frac{N_i}{I}\beta_{i,k}\exp\big(\alpha_i+\varepsilon_{i,j}\big)\right) \\
&= \gamma_j + \log\left(\overline{N}\sum_{k=1}^{K}\mu_{k,j}\left(\frac{1}{I}\sum_{i=1}^{I}\frac{N_i}{\overline{N}}\beta_{i,k}\exp\big(\alpha_i+\varepsilon_{i,j}\big)\right)\right) \\
&= \gamma_j + \log\left(\overline{N}\sum_{k=1}^{K}\mu_{k,j}B_{k,j}\right),
\end{aligned}$$

where

$$\overline{N} \overset{\text{def}}{=} \frac{1}{I}\sum_{i=1}^{I} N_i \quad \text{and} \quad B_{k,j} \overset{\text{def}}{=} \frac{1}{I}\sum_{i=1}^{I} \frac{N_i}{\overline{N}} \beta_{i,k} \exp(\alpha_i + \varepsilon_{i,j}).$$

$\mathsf{B}_{k,j}$ has some desirable properties. In particular, we have

$$\begin{aligned}
\mathbb{E}_{\hat{p}_{\mathsf{B}_{k,j}}}\left[B_{k,j}\right] &= \mathbb{E}_{\hat{p}_{\mathsf{N},\varepsilon_j,\beta_k,\alpha}}\left[\frac{N}{\mathbb{E}_i[N]}\beta_k \exp(\alpha+\varepsilon_j)\right] \\
&= \mathbb{E}_{\hat{p}_{\varepsilon_j,\beta_k,\alpha}}\left[\beta_k \exp(\alpha+\varepsilon_j)\right] \\
&= \overline{\beta}_k \mathbb{E}_{\hat{p}_{\varepsilon_j,\alpha}}\left[\exp(\alpha+\varepsilon_j)\right] \\
&\overset{\text{def}}{=} \overline{\beta}_k \beta_0,
\end{aligned}$$

and as $I \to \infty$, $\mathrm{Var}[B_{k,j}] \to 0$ (by Chebyshev's Inequality). Therefore, as $I \to \infty$ we get $B_{k,j} \approx \overline{\beta}_k \beta_0$ so

$$\log \mathbb{E}_{\mathsf{M}_j|\vec{\lambda}_j}\left[M_j \,\middle|\, \lambda_{1,j},\ldots,\lambda_{I,j}\right] \approx \gamma_j + \log\beta_0 + \log\left(\overline{N}\sum_{k=1}^{K}\mu_{k,j}\overline{\beta}_k\right).$$

Now we can estimate the platform effects γ_j. Let $W_k \overset{\text{def}}{=} \overline{\beta}_k \beta_0$. The idea is that we can approximate a model for M by

$$\begin{aligned}
\mathsf{M}_j \mid \gamma_j &\sim \text{Poisson}\left(I\overline{N}e^{\gamma_j}\sum_{k=1}^{K}\mu_{k,j}W_k\right) \\
\gamma_j &\sim \mathcal{N}(0,\sigma_\gamma^2).
\end{aligned}$$

We can then find the MLE estimates of W and σ_γ according to this model and hence obtain $\hat{\gamma}$. This is okay because conditioning on W_k, the distribution of γ_j behaves like a delta function so we can approximate γ_j by the MLE of γ.

4. **Robust Cell Type Decomposition**.

   With $\hat{\mu}_{k,j}$ and $\hat{\gamma}_j$ determined, we then find the MLE estimate for α_i, $\vec{\beta}_i$ and σ_ε in our original model.

We are not yet done after estimating each $\hat{\beta}_{i,k}$. We want to incorporate our prior information that each pixel i is a mixture of a small number of cell types. Hence, we consider models that only consider one or two cell types, and favor "singlet" models $\mathcal{M}$ by minimizing the Akaike Information Criterion

$$\mathrm{AIC}(\mathcal{M}) = \mathcal{L}(\mathcal{M}) + V \cdot p(\mathcal{M}),$$

where $\mathcal{L}$ is the log likelihood, V is a penalty parameter and $p(\cdot)$ is the number of parameters of the model.