Zero-inflated models for single-cell RNA-seq data

The ZINB team

13 juin 2016

Introduction

Single cell RNA-sequencing (scRNA-seq) is a powerful and relatively young technique to characterize molecular states of individual cells through their transcriptional profiles. It represents a major advance with respect to the standard RNA-sequencing which is only capable of detecting gene expressions averaged over millions of cells. Such averaged gene expression profiles may describe and characterize the global state of the tissue but cannot afford to study its heterogeneity and completely mask signals coming from individual cells. Accessing cell-to-cell variability is crucial for understanding many important biological processes such as tissue development and cancer. To be continued…

Model

Let Y_{ij} be the observed read count for gene $j = 1, \dots, J$ in cell $i = 1, \dots, n$, and $Z_{ij} \in \{0, 1\}$ an unobserved detection indicator, such that $Z_{ij} = 1$ when gene j is not detected in sample i (i.e., technical zero inflation) and 0 otherwise. We consider a general zero-inflated negative binomial (ZINB) model to account for zero inflation and over-dispersion of the observed counts:

$$\Pr(Y_{ij} = y) = \pi_{ij} f_0(y) + (1 - \pi_{ij}) f(y; \mu_{ij}, \phi_{ij}) \, ,$$

where π_{ij} denotes the zero-inflation (ZI) probability, $f_0(\cdot)$ the probability mass function (p.m.f.) of the count distribution when the gene is not detected (typically, the Dirac function at 0), $f(\cdot\,; \mu, \phi)$ the negative binomial (NB) p.m.f. with mean μ and dispersion parameter ϕ, i.e.,

$$f(y; \mu, \phi) = \frac{\Gamma(y + \phi^{-1})}{\Gamma(y+1)\Gamma(\phi^{-1})} \left(\frac{1}{1 + \mu\phi} \right)^{\phi^{-1}} \left(\frac{\mu}{\mu + \phi^{-1}} \right)^{y} .$$

Note that another parametrization of the NB cmf is in terms of the inverse dispersion parameter $\theta = \phi^{-1}$ (sometimes also called dispersion parameter in the literature), i.e.,

$$f(y; \mu, \theta) = \frac{\Gamma(y + \theta)}{\Gamma(y+1)\Gamma(\theta)} \left(\frac{\theta}{\theta + \mu} \right)^{\theta} \left(\frac{\mu}{\mu + \theta} \right)^{y} .$$

In both cases, the mean of the NB distribution is μ, and its variance is:

$$\sigma^2 = \mu + \phi\mu^2 = \mu + \frac{\mu^2}{\theta} \, ,$$

and the NB distribution boils down to a Poisson distribution when $\phi = 0 \Leftrightarrow \theta = +\infty$.

The mean, ZI probability and dispersion parameter for the count of the gene j in cell i are modeled as follows:

$$\log(\mu_{ij}) = \left(X\beta_\mu + U\gamma_\mu + (V\delta_\mu)^\top + W\alpha_\mu \right)_{ij} \, , \tag{1}$$

$$\text{logit}(\pi_{ij}) = \left(X\beta_\pi + U\gamma_\pi + (V\delta_\pi)^\top + W\alpha_\pi \right)_{ij} \, , \tag{2}$$

$$\log(\phi_{ij}) = \phi_j \, , \tag{3}$$

where

- X is an $n \times M$ design matrix corresponding to M covariates of interest/factors of "wanted variation" (e.g., cell type) and $\beta = (\beta_\mu, \beta_\pi)$ its associated M × J matrices of parameters of interest;
- U is an $n \times L$ matrix corresponding to known sample-level unwanted factors (e.g., C1 run, QC measures) and $\gamma = (\gamma_\mu, \gamma_\pi)$ its associated L × J matrices of nuisance parameters;
- V is a $J \times L$ matrix corresponding to known gene-level unwanted factors (e.g., GC-content) and $\delta = (\delta_\mu, \delta_\pi)$ its associated $L \times n$ matrices of nuisance parameters;
- W is an unobserved $n \times K$ matrix corresponding to unknown factors, which could be of "unwanted variation" as in RUV or of interest (such as biological variations), and $\alpha = (\alpha_\mu, \alpha_\pi)$ its associated $K \times J$ matrices of parameters.

This model deserves a few comments

- The model extends the RUV framework to ZINB variables. It differs in interpretation from RUV in the $W\alpha$ factor, which we do not consider as necessarily unwanted; it generally corresponds to an unknown low-dimensional variations in the data, that could be due to unwanted factors such as technical artefacts (as in RUV), or to biological variations which could be of interest such as cell cycle or cell differentiation, as typically assumed in other factor models such as PCA or ICA.
- By default U and V always contain at least one constant column, to account for cell-specific (a.k.a. size factors) and gene-specific (e.g., mean expression level) variations.
- The X, U and V matrices could differ in the modelling of μ and π if we assume that some known factors do not affect all three parameters; to keep notations simple and consistant we use the same matrices, but will implicitly assume that some parameters may be constrained to be 0 if needed.
- When $X = 0$, $U = \mathbf{1}_n$ and $V = \mathbf{1}_J$, then the model if a factor model akin to PCA where W is a factor matrix and (α_μ, α_π) are loading matrices.
- By allowing the parameters to differ between the models of μ and π, we can model and test for differences in NB mean or in ZI probability.
- We limit ourselves to a gene-dependent dispersion parameter. More complicated models for ϕ_{ij} could be investigated, such as a model similar to μ_{ij}, or a functional of the form $\phi_{ij} = f(\mu_{ij})$, but we restrict ourselves to a simpler model that has been shown to be largely sufficient in bulk RNA-seq models.

Parameter estimation

The input to the model are the matrices X, U, V and the integer K; the parameters to be inferred β, γ, δ, W, α and ϕ. The log-likelihood of the model is

$$\begin{aligned} \ell(\beta, \gamma, \delta, W, \alpha, \phi) &= \sum_{i=1}^{n} \sum_{j=1}^{J} \ln \Pr(Y_{i,j} = y_{ij} \,|\, \beta, \gamma, \delta, W, \alpha, \phi) \\ &= \sum_{i=1}^{n} \sum_{j=1}^{J} \ln \left(\pi_{ij} f_0(y_{ij}) + (1 - \pi_{ij}) f(y_{ij}; \mu_{ij}, \phi_{ij}) \right) \end{aligned} \tag{4}$$

To infer the parameters we follow a penalized maximum likelihood approach, by trying to solve

$$\max_{\beta,\gamma,\delta,W,\alpha,\phi} \left\{ \ell(\beta, \gamma, \delta, W, \alpha, \phi) - \epsilon \text{Pen}(\beta, \gamma, \delta, W, \alpha, \phi) \right\} ,$$

where $(Pen)(\cdot)$ aims at reducing overfitting and improving numerical stability of the optimization problem, when many parameters must be estimated. By default we take

$$\mathrm{Pen}(\beta, \gamma, \delta, W, \alpha, \phi) = \|\beta^0\|^2 + \|\gamma^0\|^2 + \|\delta\|^2 + \|W\|^2 + \|\alpha\|^2 + \mathrm{var}(\phi)^2 \,,$$

where β^0 and γ^0 stand for the matrices β and γ without the columns corresponding to the constant covariate, and $\|\cdot\|$ is the Frobenius matrix norm. The penalty tends to shrink the estimated parameters to 0, except for the cell- and gene-specific effects which are not penalized, and the for the dispersion parameters which are not shrinked towards 0 but instead towards a constant value across genes. Note also that

$$\min_{W,\alpha \,:\, W\alpha = R} \|W\|^2 + \|\alpha\|^2 = \|R\|_* \,,$$

where $\|R\|_*$ is the trace norm (a.k.a. nuclear norm), i.e., the sum of singular values of R, and that the minimum is reached when W and α form orthogonal bases of left- and right-singular vectors; in particular, this implies that this penalty will enforce orthogonal columns for W and α, which is useful for visualization or interpretation of latent factors.

The penalized likelihood is however not concave, making its maximization computationally challenging. We instead find a local maximum starting from a smart initialization and iterating a numerical optimization scheme until local convergence, as described below.

Initialization

To initialize the set of parameters we approximate the count distribution by a log-normal distribution and explicitly separate zero and non-zero values, as follows:

1. Set $\mathcal{P} = \{(i,j) \,:\, Y_{ij} > 0\}$
2. Set $L_{ij} = \log(y_{ij})$ for all $(i,j) \in \mathcal{P}$.
3. Estimate $\hat{\delta}_\mu$ by solving the ridge regression problem:

   $$\hat{\delta}_\mu \in \arg\min_{\delta_\mu} \sum_{(i,j)\in\mathcal{P}} \left(L_{ij} - (V\delta_\mu)_{ji}\right)^2 + \epsilon\left(\|\delta_\mu^0\|^2\right) \,.$$

   Note that this problem can be solved for each cell (i) independently and in parallel. When there is no gene covariate besides the constant offset, the problem is easily solved by setting $(\hat{\delta}_\mu)_i$ to the mean of the log-counts (for non-zero counts) of the cell i.
4. Remove the cell-specific estimates from L by setting

   $$\forall (i,j) \in \mathcal{P}, \quad L_{ij} \leftarrow L_{ij} - (V\hat{\delta}_\mu)_{ji} \,.$$

5. Estimate $\hat{\beta}_\mu$ and $\hat{\gamma}_\mu$ by solving a ridge regression problem:

   $$\left(\hat{\beta}_\mu, \hat{\gamma}_\mu\right) \in \arg\min_{(\beta_\mu, \gamma_\mu)} \sum_{(i,j)\in\mathcal{P}} \left(L_{ij} - (X\beta_\mu + U\gamma_\mu)_{ij}\right)^2 + \epsilon\left(\|\beta_\mu\|^2 + \|\gamma_\mu^0\|^2\right) \,.$$

   Note that this problem can be solved for each gene (j) independently and in parallel. Again, when there is no cell-specific covariate besides the constand offset then the gene-specific parameter $(\gamma_\mu)_j$ is simply obtained by the mean of L_{ij} across cells.
6. Remove the gene-specific estimates from L by setting

   $$\forall (i,j) \in \mathcal{P}, \quad L_{ij} \leftarrow L_{ij} - (X\hat{\beta}_\mu + U\hat{\gamma}_\mu)_{ji} \,.$$

7. Estimate $\hat{W}$ and $\hat{\alpha}$ by performing a SVD with missing values from the L matrix (with entries restricted to $\mathcal{P}$). Here we use a technique similar to the the *SVDImpute()* function of the *imputation* package, which starts by filling missing values in the L matrix using the mean of the columns, then computes a low-rank approximation of the matrix by rank-M SVD, then fill the missing values again from the rank-M approximation, and iterates a few times. Other techniques may be used a well, such as stochastic gradient-based approaches popular in recommender systems. Once a rank-M SVD of L is obtaines, of the form $L \approx FDG^\top$, we set $\hat{W} = D^{\frac{1}{2}}F$ and $\hat{\alpha} = D^{\frac{1}{2}}G^\top$.

8. Set $\hat{Z}_{ij} = 1$ if $(i,j) \in \mathcal{P}$, $\hat{Z}_{ij} = 0$ otherwise.

9. Estimate $\hat{\delta}_\pi$ by solving the regularized logistic regression problem:

   $$\hat{\delta}_\pi \in \arg\min_{\delta_\pi} \sum_{(i,j)} \left[-\hat{Z}_{ij}(V\delta_\pi)_{ji} + \log\left(1 + e^{(V\delta_\pi)_{ji}}\right) \right] + \epsilon\left(\|\delta_\pi^0\|^2\right) \,.$$

   Note that this problem can be solved for each cell (i) independently and in parallel. When there is no gene covariate besides the constant offset, the problem is easily solved by setting $(\hat{\delta}_\mu)_i$ to the logit of the proportion of zeros in each cell.

10. Estimate $\hat{\beta_\pi}$, $\hat{\gamma}_\pi$ and $\hat{\alpha}_\pi$ by solving the regularized logistic problem

    $$\begin{aligned}\left(\hat{\beta}_\pi, \hat{\gamma}_\pi, \hat{\alpha}_\pi\right) \in \arg \min_{(\beta_\pi,\gamma_\pi,\alpha_\pi)} \sum_{(i,j)} \Big[& - \hat{Z}_{ij}(X\beta_\pi + U\gamma_\pi + (V\hat{\delta}_\pi)^\top + W\alpha_\pi)_{ij} \\ & + \log\left(1 + e^{(X\beta_\pi + U\gamma_\pi + (V\hat{\delta}_\pi)^\top + W\alpha_\pi)_{ij}}\right) \Big] + \epsilon\left(\|\beta_\pi\|^2 + \|\gamma_\pi^0\|^2 + \|\alpha_\pi\|^2\right) \,. \qquad (5)\end{aligned}$$

11. Initialize $\hat{\phi} = 0$ (no overdispersion).

Optimization

After initialization, we maximize locally the penalized log-likelihood by alternating left- and right-factor optimization, iterating the following steps until convergence:

1. Left-factor (cell-specific) optimization:

   $$\left(\hat{\delta}, \hat{W}\right) \leftarrow \arg\max_{(\delta,W)} \left\{ \ell(\hat{\beta}, \hat{\gamma}, \delta, W, \hat{\alpha}, \hat{\phi}) - \epsilon \mathrm{Pen}(\hat{\beta}, \hat{\gamma}, \delta, W, \hat{\alpha}, \hat{\phi}) \right\} \,,$$

   Note that this optimization can be performed independently and in parallel for each cell i.

2. Right-factor (gene-specific) optimization:

   $$\left(\hat{\beta}, \hat{\gamma}, \hat{\alpha}, \hat{\phi}\right) \leftarrow \arg \max_{(\beta,\gamma,\alpha,\phi)} \left\{ \ell(\beta, \gamma, \hat{\delta}, \hat{W}, \alpha, \phi) - \epsilon \mathrm{Pen}(\beta, \gamma, \hat{\delta}, \hat{W}, \alpha, \phi) \right\} \,,$$

   Note that this optimization can be performed independently and in parallel for each gene j.

3. Orthogonalization:

   $$\left(\hat{W}, \hat{\alpha}\right) \leftarrow \arg \min_{(W,\alpha)\,:\,W\alpha = \hat{W}\hat{\alpha}} \left(\|W\|^2 + \|\alpha\|^2\right) \,.$$

   This is obtained by performing a SDV of $\hat{W}\hat{\alpha} = FDG^\top$, and setting $\hat{W} \leftarrow FD^{\frac{1}{2}}$ and $\hat{\alpha} \leftarrow D^{\frac{1}{2}}G^\top$. Note that this step not only allows to maximize locally the penalized log-likelihood, but also ensures that the columns of W stay orthogonal to each other during optimization.

TODO: write gradients and explain how steps 1 and 2 are implemented using a single function