Isolating the perturbation response of gene regulatory networks in the presence of biological variability and technical noise

Clayton W. Seitz

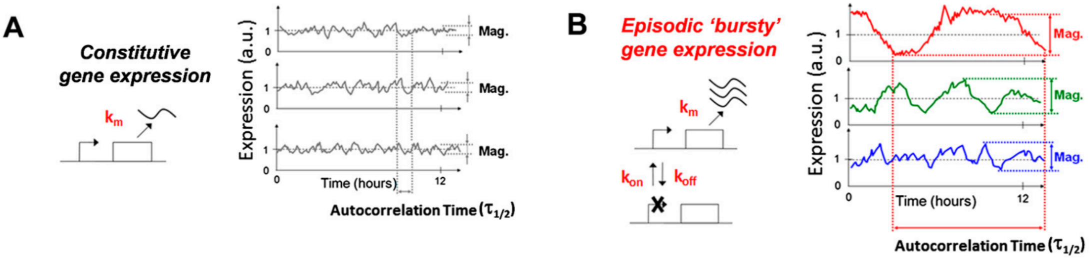
May 12, 2022

Outline

Modeling stochastic biochemical reaction networks

Application to chemotherapy resistance in melanoma

Gene expression can be non-constitutive


A
Constitutive gene expression
k_m
Expression (a.u.)
1
0
Mag.
Time (hours)
0
12
Autocorrelation Time ($\tau_{1/2}$)
B
Episodic 'bursty' gene expression
k_m
k_{on}
k_{off}
Expression (a.u.)
1
0
Mag.
Time (hours)
0
12
Autocorrelation Time ($\tau_{1/2}$)

- Non-equilibrium expression cannot be measured directly with *ensemble snapshots*
- Tracking counts in live cells limits the number of species considered simultaneously
- If the biochemical network is known a-priori, we can build parametric dynamical models
- Bayesian inference allows us to fit parametric dynamical models without continuous dynamical trajectories

Stochastic biochemical reaction networks: the repressilator

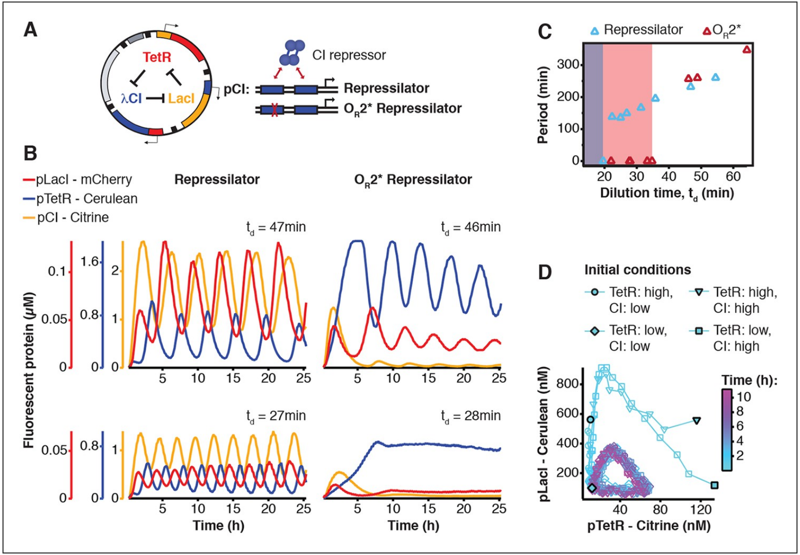


Figure 1: Niederholtmyer et al., eLife 2015

Bayesian parameter inference for gene regulation

Suppose we have a series of ensemble snapshots of an *in-vitro* population:

$$\mathcal{D} = \{\mathcal{D}^0, ..., \mathcal{D}^t\}$$

with $\mathcal{D}^t = \{d_1, ..., d_n\}$. We would like to use $\mathcal{D}$ to fit a dynamical model $\mathcal{M}(\theta)$

Bayesian inference lets us infer θ from $\mathcal{D}$ while quantifying the uncertainty in our estimate:

$$P(\theta|\mathcal{D}) \propto P(\mathcal{D}|\theta)P(\theta) = P(\theta)\prod_{n,t} P(\mathcal{D}_n^t|\theta)$$

The likelihood $P(\mathcal{D}_n^t|\theta)$ is often difficult to define or intractable to compute due to the curse of dimensionality

Estimating the likelihood

If all reactions take place in a well-mixed solution (Markov), the chemical master equation (CME) applies

$$\frac{dP(\mathbf{x},t)}{dt} = \sum_j a_j(\mathbf{x}-\nu_j|\theta)P(\mathbf{x}-\nu_j,t) - a_j(\mathbf{x}|\theta)P(\mathbf{x},t)$$

$P(\mathcal{D}^t|\theta)$ is the solution to the master equation $P(\mathbf{x},t)$ under parameterization θ

ABC methods simulate data $\tilde{\mathcal{D}}$ from $\mathcal{M}(\theta)$ using the Gillespie algorithm and compute a distance metric $d(\mathcal{D},\tilde{\mathcal{D}})$ or $d(\mathcal{S}(\mathcal{D}),\mathcal{S}(\tilde{\mathcal{D}}))$ where $\mathcal{S}$ is a summary statistic to approximate the likelihood

Both suffer from the curse of dimensionality

Beating the curse of dimensionality to compute the likelihood

Deep networks are known to be capable of circumventing the curse. They can model very high-dimensional joint distributions e.g., distributions over images

Can we use a deep network to perform the map $\Theta \in \mathbb{R}^n \rightarrow f(\Theta) = P(\mathcal{D}|\Theta) \in \mathbb{R}^{n+1}$ i.e., compute the likelihood?

Drug-induced reprogramming as a mode of cancer drug resistance

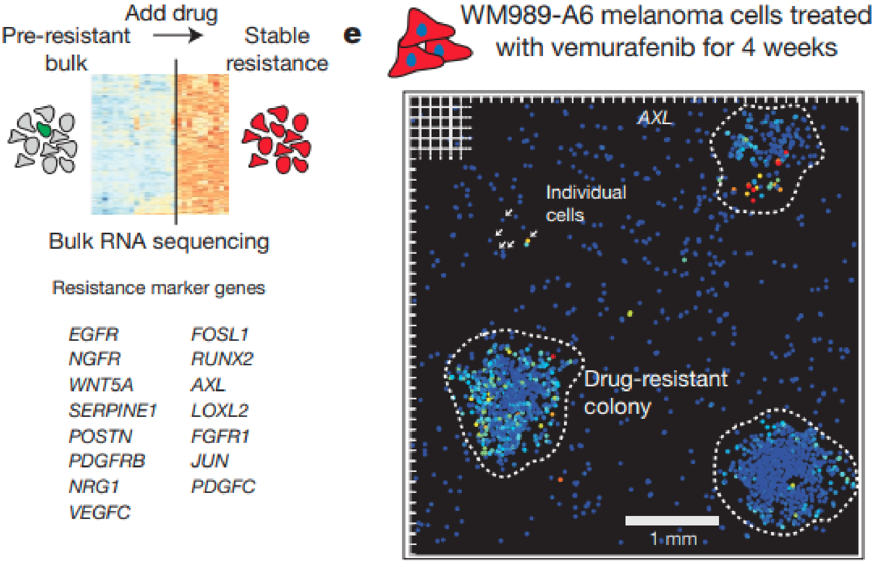
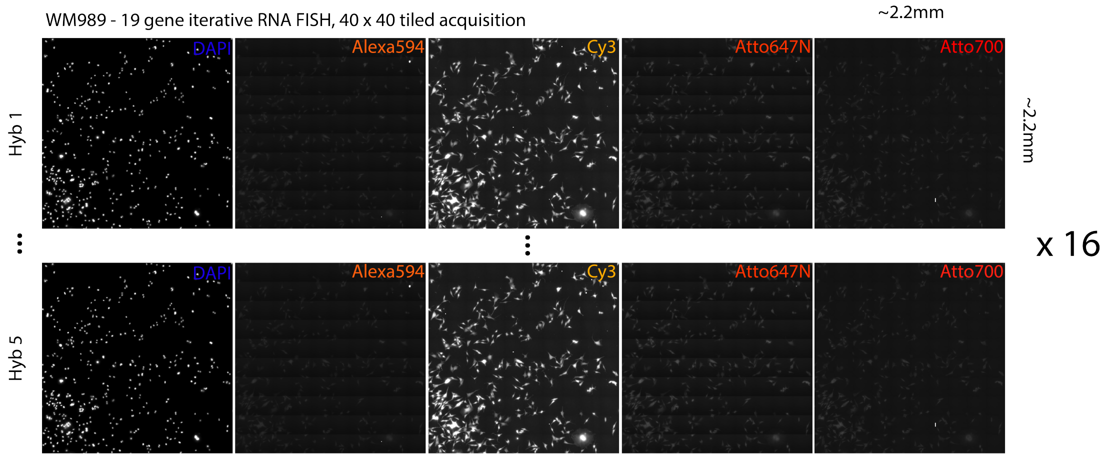


Figure 2: Shaffer et al., Nature 2017

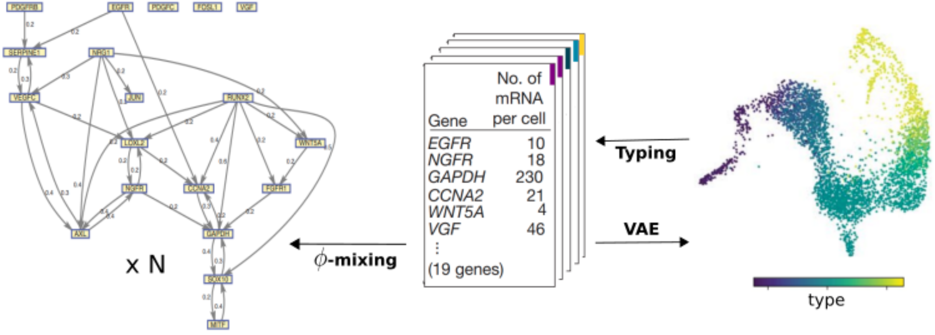
WM989-A6 RNA-FISH data summary

WM989 - 19 gene iterative RNA FISH, 40 x 40 tiled acquisition



Hyb 1	EGFR	SOX10	CCNA2	GAPDH
Hyb 2	WNT5A	PDGFRB	PDGFC	SERPINE1
Hyb 3	NGFR	NRG1	FOSL1	VEGFC
Hyb 4	AXL	MITF	LOXL2	RUNX2
Hyb 5	FGFR1	JUN	VGF	

Graphical abstract

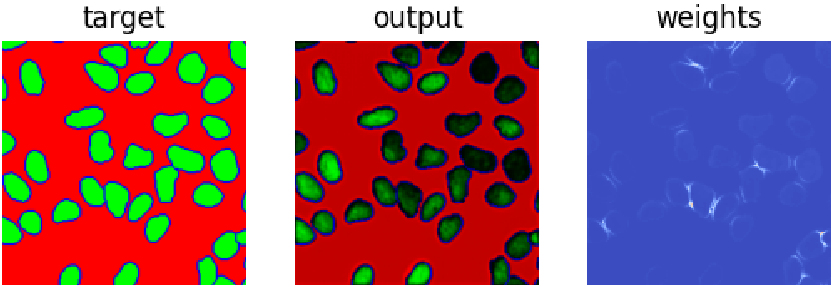


Gene	No. of mRNA per cell
EGFR	10
NGFR	18
GAPDH	230
CCNA2	21
WNT5A	4
VGF	46
⋮	
(19 genes)	

Figure 3: *Gene expression matrices used as training data to learn a latent-space representation of gene expression, uncovering latent structure of the joint distribution and permitting cell typing, account for batch variability. Type information is then used for inference of the underlying regulatory network using the phi-mixing coefficient, which may differ across types*

Training on BBBC039 U2OS Cells

BBBC039: 200 images, 160 train + 40 validation, 256 x 256 random crop

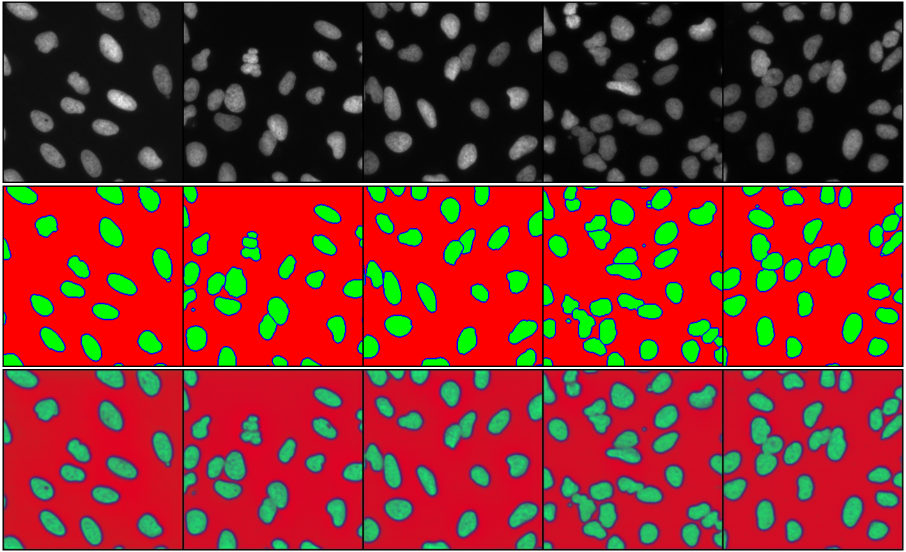


We train a 3-channel semantic segmentation model with **weighted** cross-entropy loss:

$$\mathcal{L} = \sum_{i,j} w_{ij} \log p_{ij}(\tilde{x}) = \sum_{i,j} w_{ij} \log \frac{\exp(-s_{ij}(\tilde{x}))}{\sum_{x \in \chi} \exp(-s_{ij}(\tilde{x}))}$$

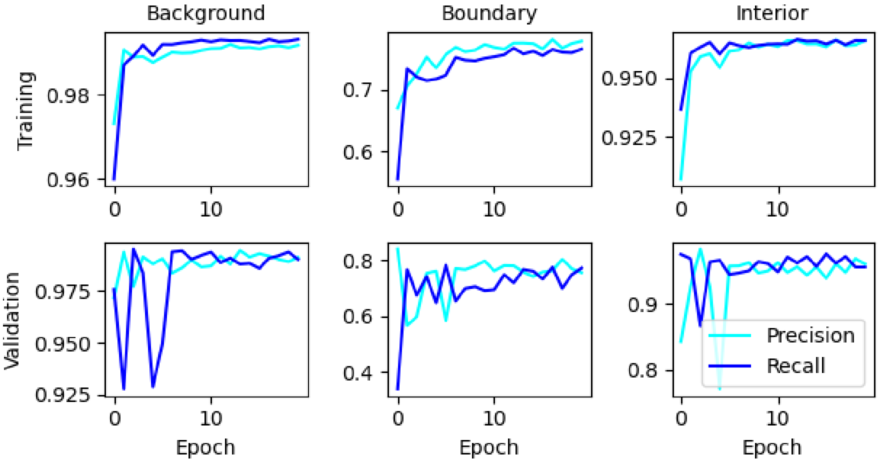
p_{ij} is the probability the model assigns a pixel to the true class $\tilde{x} \in \{\mathrm{a}, \mathrm{b}, \mathrm{c}\}$

Training on BBBC039 U2OS Cells



Training on BBBC039 U2OS Cells

Learning rate $\eta = 0.01$, Batch-size $B = 5$ (32 train iterations, 8 validation)



References I