

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

Clayton W. Seitz

May 12, 2022

Outline

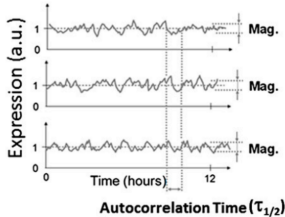
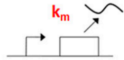
Modeling stochastic biochemical reaction networks

Application to chemotherapy resistance in melanoma

Gene expression can be non-constitutive

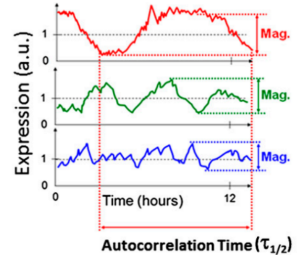
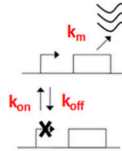
A

**Constitutive
gene expression**



B

**Episodic 'bursty'
gene expression**



- ▶ 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

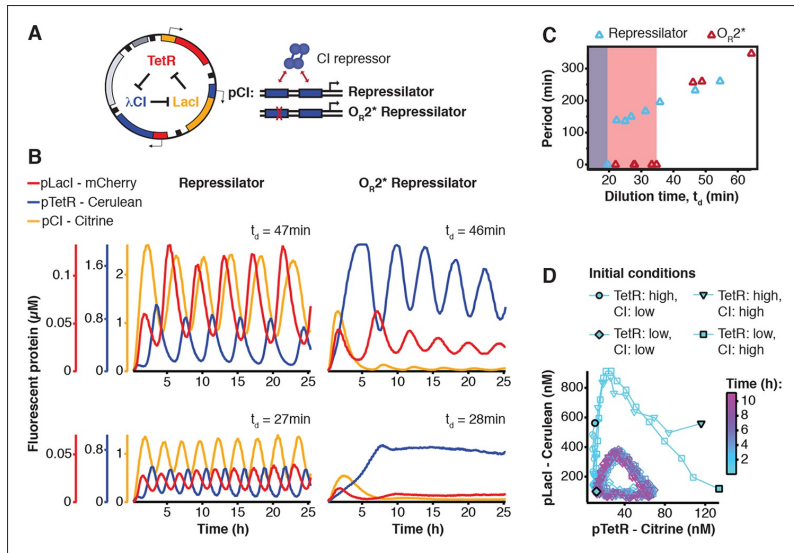


Figure 1: Niederholtmyer et al., eLife 2015

What about Markov models?

From another point of view, molecular counts can be thought of as a Markov model which progresses through states



Markov models are very powerful, and the EM algorithm can be used for MAP estimation, but this requires time-series measurements, which are incompatible with ensemble snapshots

Bayesian parameter inference for gene regulation

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

$$\mathbf{x} = \{\mathbf{x}^0, \dots, \mathbf{x}^t\} \quad \mathbf{y} = \{\mathbf{y}^0, \dots, \mathbf{y}^t\}$$

with $\mathbf{x}^t = \{x_1, \dots, x_n\}$ and similarly for \mathbf{y} . Under perfect measurements $\mathbf{x} = \mathbf{y}$

We would like to use \mathbf{x} to fit a dynamical model $\mathcal{M}(\theta)$. Bayesian inference lets us infer θ from \mathbf{x} while quantifying the uncertainty in our estimate:

$$P(\theta|\mathbf{x}) \propto \mathcal{L}(\mathbf{x}|\theta)P(\theta) = P(\theta) \prod_t P(\mathbf{x}, t|\theta)$$

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

The Chapman-Kolmogorov Equation

Consider the discrete-time Chapman-Kolmogorov equation for a discrete state space

$$\begin{aligned} P(\mathbf{x}_i, t) &= \sum_j T_{ji}(\mathbf{x}_i, t | \mathbf{x}_j, t - \Delta t) P(\mathbf{x}_j, t - \Delta t) \\ &= \sum_k T_k(\mathbf{x}_i, t | \mathbf{x}_i - \nu_k, t - \Delta t) P(\mathbf{x}_i - \nu_k, t - \Delta t) \end{aligned}$$

where T_k is the probability of a reaction channel k firing in the interval $(t, t + \Delta t)$.

Taking the limit $\Delta t \rightarrow 0$ one can derive the forward Kolmogorov equation or chemical master equation (CME)

$$\frac{dP(\mathbf{x}, t | \mathbf{x}_0)}{dt} = \sum_k T_k(\mathbf{x} - \nu_k) P(\mathbf{x} - \nu_k, t) - T_k(\mathbf{x}) P(\mathbf{x}, t)$$

Estimating the likelihood function

The finite state projection (FSP) offers a 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

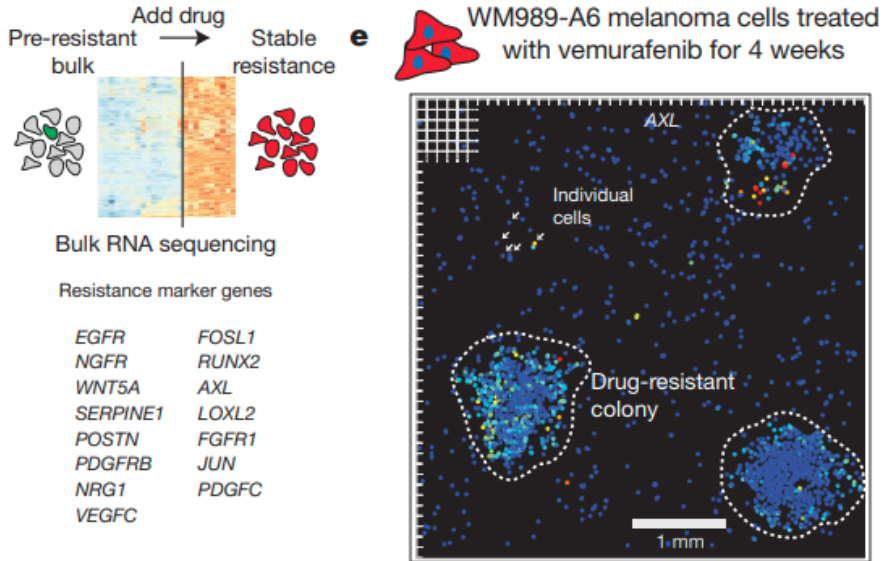
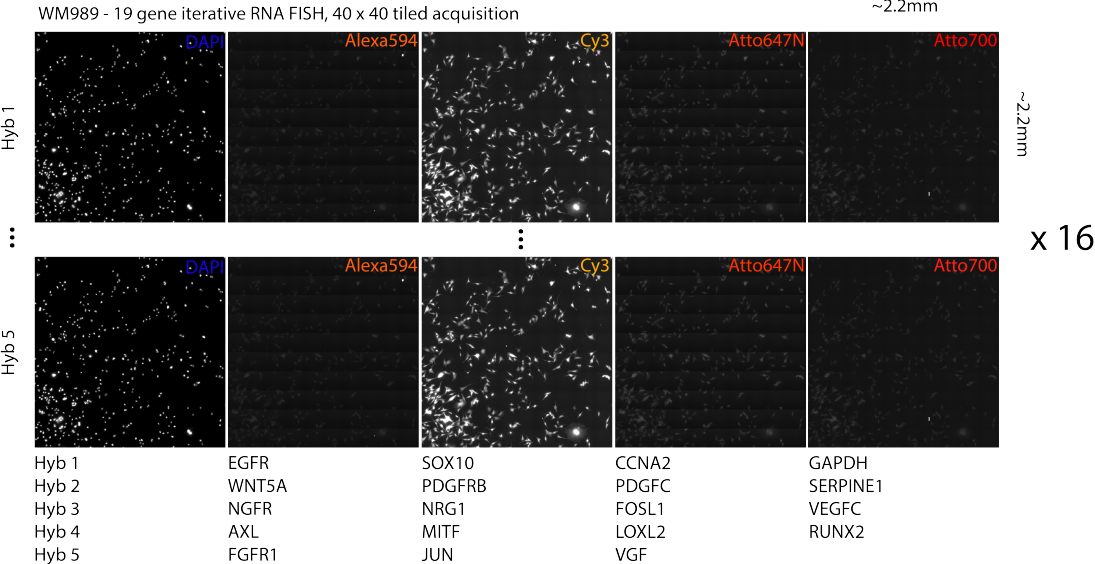


Figure 2: Shaffer et al., Nature 2017

WM989-A6 RNA-FISH data summary



Graphical abstract

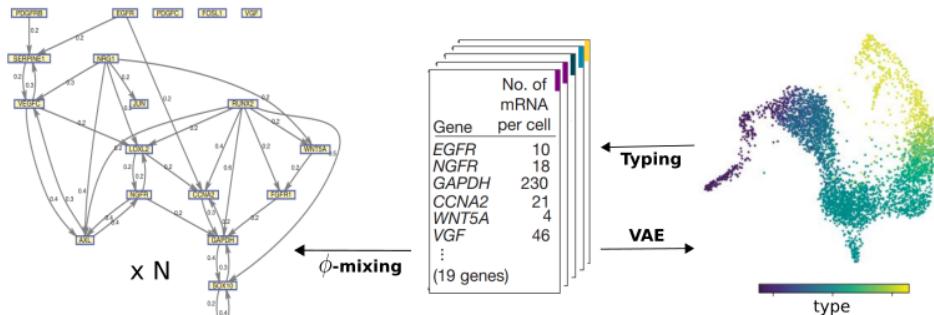
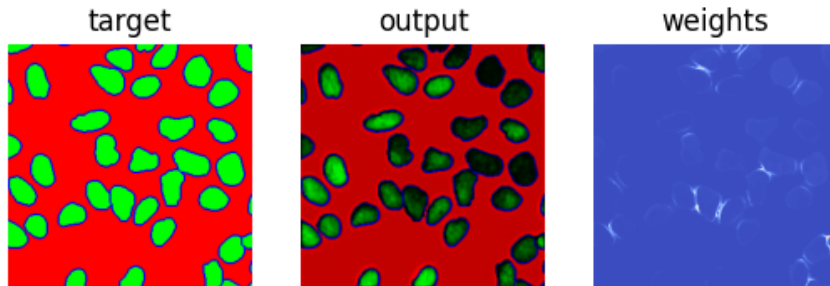


Figure 3: Gene expression matrices used as training data to learn a latent-space representation of gene expression, uncovering latent structure of the joint distribution and permitting cell typing, account for batch variability. Type information is then used for inference of the underlying regulatory 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

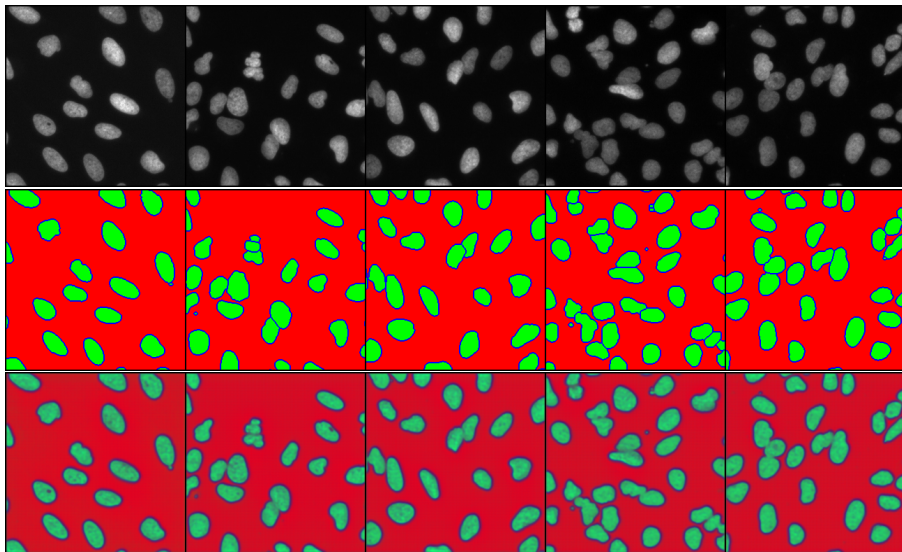


We train a 3-channel semantic segmentation model with **weighted** cross-entropy loss:

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

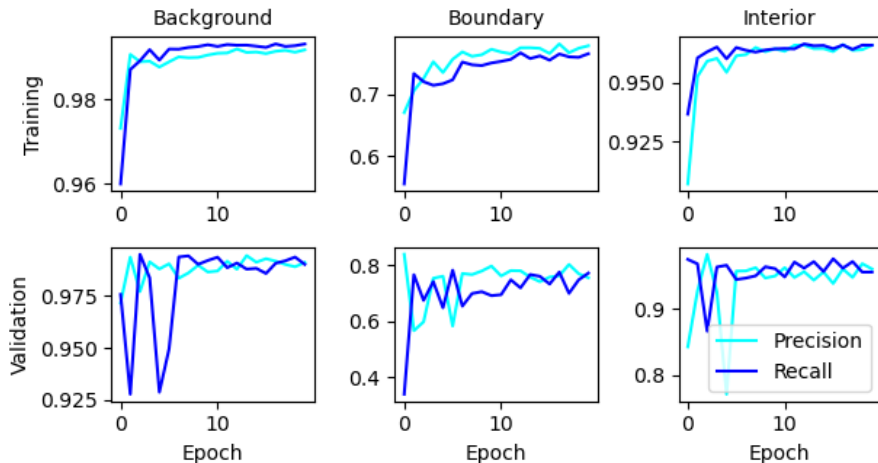
p_{ij} is the probability the model assigns a pixel to the true class $\tilde{x} \in \{a, b, c\}$

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