Inference and simulation of gene regulatory networks.

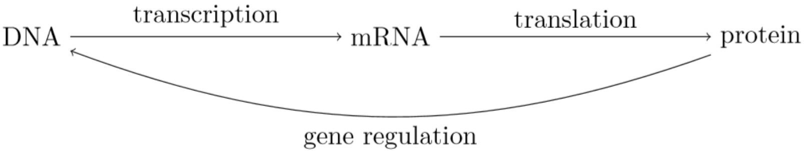
Elias Ventre

Thibault Espinasse, Thomas Lepoutre,
Olivier Gandrillon, Ulysse Herbach.

July, 07th 2022

Introduction

- Gene expression is the process by which its **DNA** is transcripted and then translated into **proteins**. The protein produced by a gene can impact the transcription of the other genes.



- We consider a cell $X_t = (X_{1,t}, \cdots, X_{n,t})$ evolving in the **gene expression space:** for each gene i, $X_{i,t} = (m_i, p_i)_t$.

Introduction

- Differentiation is a **stochastic process**: we consider that the variability stems from transcriptional level, by the **regulatory effects** of proteins on the transcription of the other genes.

- The evolution of a cell depends on its **Gene Regulatory Network** (GRN):

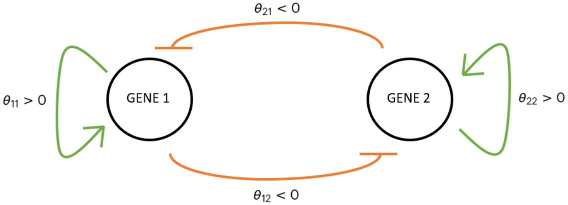


Figure: GRN example - the toggle-switch

Inference from expression data

- **Non-parametric methods**: We do not make hypothesis on the nature of gene regulation, just for deducing information from data:

→ *Correlation coefficients, Information theoretic score (mutual information), Tree-based ensemble method...*

- **Parametric methods**: We make hypothesis on a theoretic procedure that explains the data from a set of parameters Θ:

→ *Statistical model, Deterministic dynamical model, Boolean networks,* **Stochastic dynamical model***...*

Examples of models for n measurements $(M_1, \cdots, M_n)$

- **Example of parametric statistical model**:
Every $M_i \sim \mathcal{P}(Y_i)$, where the Poissonian noise is due to the measure, Y_i being the "real" mRNA concentration verifying $Y_i \sim \Gamma(\alpha, \beta)$.
$\Theta = (\alpha, \beta)$.

$\rightarrow$ It remains difficult to give a meaning to the parameters !

Examples of models for n measurements $(M_1, \cdots, M_n)$

- **Example of parametric stochastic dynamical model**:
Every M_i is a realization of the system verifying on every small interval $[t, t+\Delta t]$:

$$M(t+\Delta t) = M(t) - d\Delta_t M(t) + s1_{\mathcal{E}(a)>\Delta_t}.$$

$\Theta = (d, s, a) =$ *(degradation, creation intensity, creation frequency).*

$\rightarrow$ We need to analyse the model for interpreting the observations.

- **Remark**: If the data are not time-stamped, we have to assume that they correspond to samples of of the model after a long-time. If not, we have to know the initial conditions!

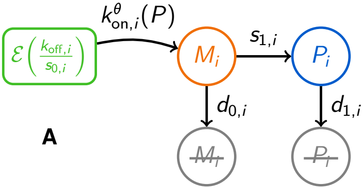
Table of Contents

1. **Mathematical model of gene expression for a parametric approach of network inference.**

2. Inference using metastability.

3. Results.

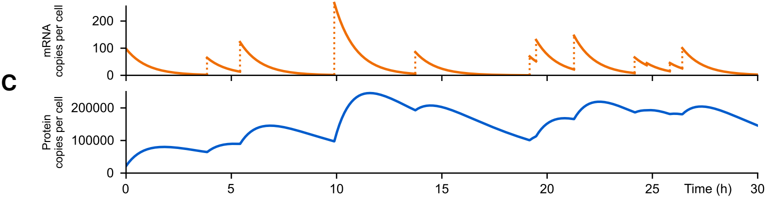
Stochastic Two States Model in a bursty regime



$\mathcal{E}\left(\frac{k_{\text{off},i}}{s_{0,i}}\right)$
$k^{\theta}_{\text{on},i}(P)$
M_i
$s_{1,i}$
P_i
$d_{0,i}$
$d_{1,i}$
$\cancel{M_i}$
$\cancel{P_i}$
A

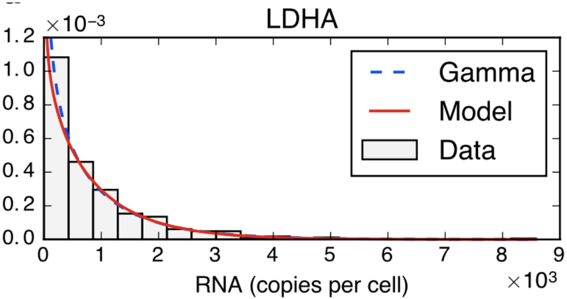
B

$$\begin{cases} M_i(t) \xrightarrow{k^{\theta}_{\text{on},i}(P(t))} M_i(t) + \mathcal{E}\left(\frac{k_{\text{off},i}}{s_{0,i}}\right) \\ M_i'(t) = -d_{0,i}M_i(t) \\ P_i'(t) = s_{1,i}M_i(t) - d_{1,i}P_i(t) \end{cases}$$



Models distribution for one gene

- The hybrid model is **analytically tractable** for constant parameters: its stationary distribution is a **Gamma distribution**.



GRN and jumps rate functions

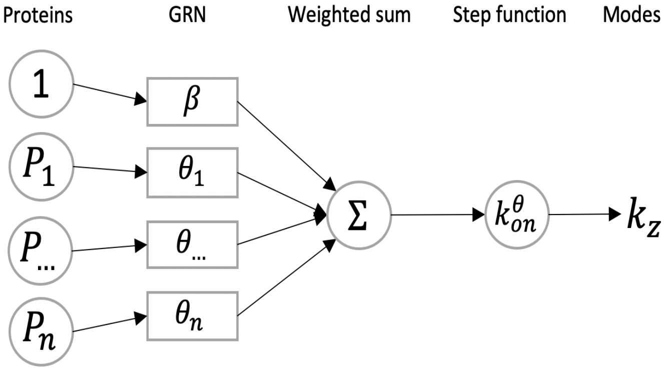
- The GRN is characterized by a matrix $\Theta \in M_n(\mathbb{R})$ which appears in the model through **the choice of the functions** $k^{\theta}_{on,i}(P)$**.**

$$k^{\theta}_{on,i}(P) = k_{0,i} + (k_{1,i} - k_{0,i})\sigma^{\theta}_i(P),$$

where $\sigma^{\theta}_i(P) = \left(1 + \exp\left(-\beta_i - \sum_{j=1}^{n} \theta_{ij} P_j\right)\right)^{-1}$.

$$\Longrightarrow \forall i = 1, \cdots, n : \begin{cases} M_i(t) \xrightarrow{\mathbf{k^{\theta}_{on,i}(P(t))}} M_i(t) + \mathcal{E}\left(\frac{k_{off,i}}{s_{m,i}}\right), \\ M_i'(t) = -d_{m,i} M_i(t), \\ P_i'(t) = s_{p,i} M_i(t) - d_{p,i} P_i(t). \end{cases}$$

Analogy with neural network



When the function $k_{on,\Theta}$ is sigmoidal, the activity of a gene can be compared as controlled by a neuron.

Existing strategy for inferring such model

- **From simulations** $\sim$ Model selection (*Koshkin et al. 2021*):

$\rightarrow$ *Main limitations*: **Difficulty** for comparing stochastic realizations, **time consuming** when there are many genes !

- **From distributions** $\sim$ Maximum likelihood (*Herbach et al. 2017*):

$\rightarrow$ *Main limitation*: the model is **too complex** for the distribution to be explicitly known with respect to θ, especially in the **non-stationary state** !

Table of Contents

1. Mathematical model of gene expression for a parametric approach of network inference.

2. **Inference using metastability.**

3. Some results.

Simplified model (temporarily)

- We consider a simplified model by skipping mRNA:

$$\forall i = 1, \cdots, n : \begin{cases} P_i(t) \xrightarrow{k^{\theta}_{on,i}(P(t))} P_i(t) + \mathcal{E}\left(c_i\right), \\ P_i'(t) = -d_i P_i(t), \end{cases}$$

where we define $c_i = \frac{k_{off,i} d_{m,i}}{s_{m,i} s_{p,i}}$.

- In that case we can consider that: $M_i | P \sim \Gamma(\frac{k^{\theta}_{on,i}(P)}{d_m}, \frac{k_{off,i}}{s_m})$.

Deterministic approximation

- We introduce a **scaling factor** ε characterizing the relative velocity of promoters switches in regard to protein dynamics:

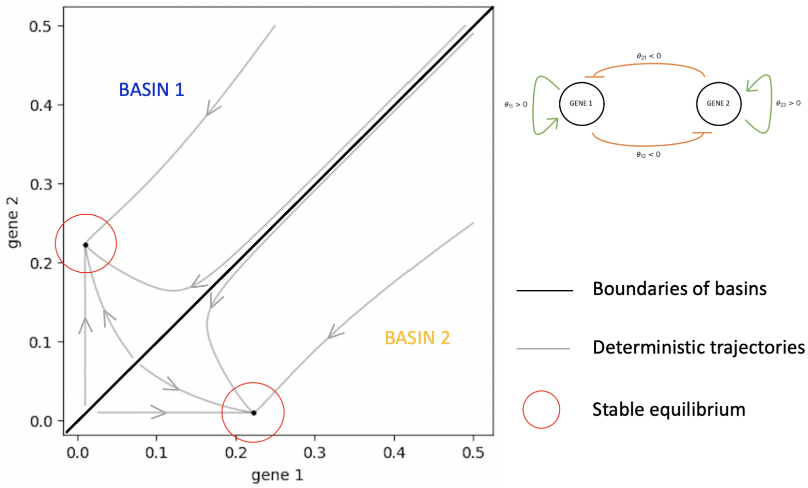
$$\varepsilon = \frac{\bar{d}}{\bar{k}},$$

- **scaling factor** $\sim$ **noise coefficient**

If $\varepsilon \ll 1$, we derive a **deterministic limit**:

$$\begin{aligned} \bar{P}'(t) &= \frac{k_{off}\, d_m}{s_m s_p} k_{on}^{\theta}\left(\bar{P}(t)\right) - d_p \bar{P}(t), \\ \rightarrow\ \bar{P}'(t) &= F^{\theta}\left(\bar{P}(t)\right). \end{aligned}$$

Deterministic limit of a toggle-switch network

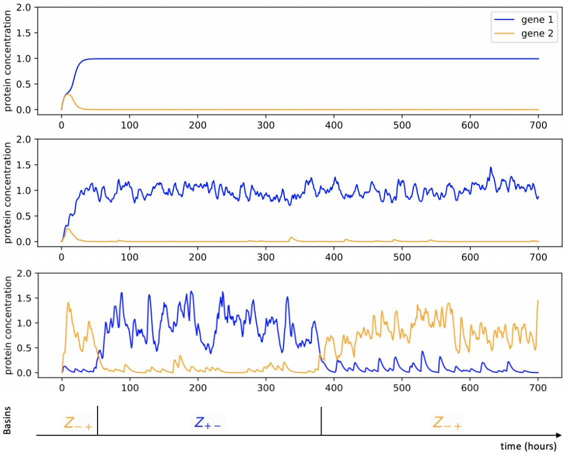


Boundaries of basins

Deterministic trajectories

Stable equilibrium

Stochastic trajectories of a toggle-switch



$\Longrightarrow$ The main behaviour of a cell is described by **the transitions between the basins**, which are seen as **cell types**.

Metastability

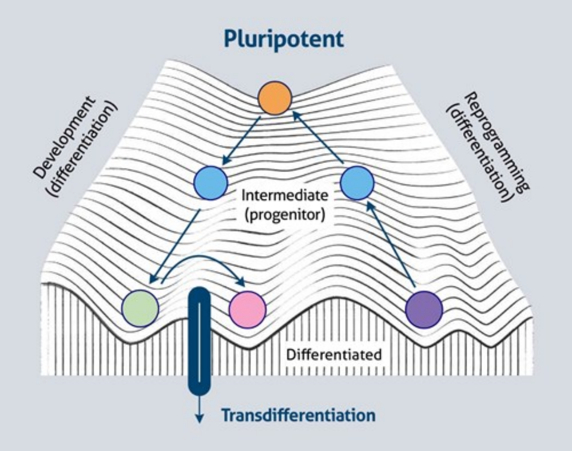
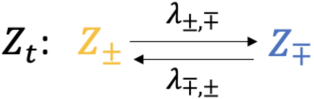

Pluripotent
Development (differentiation)
Reprogramming (differentiation)
Intermediate (progenitor)
Differentiated
Transdifferentiation

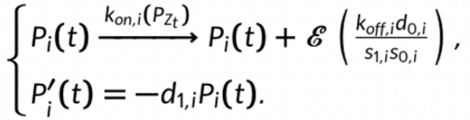
Figure: Waddington's epigenetic landscape is a metaphor for how gene regulation modulates development.

Phenomenological model

- We derive a phenomenological model which approximates the PDMP system by considering **the independence of genes knowing a basin**:



$$Z_t : \quad Z_\pm \underset{\lambda_{\mp,\pm}}{\overset{\lambda_{\pm,\mp}}{\rightleftarrows}} Z_\mp$$



$$\begin{cases} P_i(t) \xrightarrow{k_{on,i}(P_{Z_t})} P_i(t) + \mathscr{E}\left(\frac{k_{off,i}d_{0,i}}{s_{1,i}s_{0,i}}\right), \\ P_i'(t) = -d_{1,i}P_i(t). \end{cases}$$

The main idea consists in approximating within each basin z:

$$k_{on,i}(P) \simeq k_{on,i}(P_z) = k_{z,i}.$$

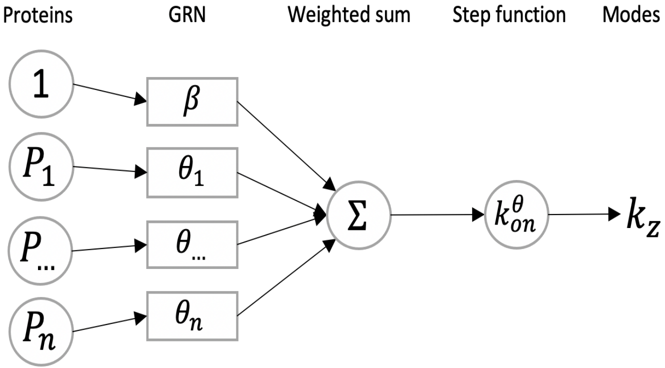
Mixture approximation

- The stationary distribution is a Gamma mixture :

$$u \sim \sum_{z \in Z} \mu_z \prod_{i=1}^{g} Gamma\left(\frac{k_{z,i}}{d_{1,i}}, \frac{k_{off,i} d_{m,i}}{s_{m,i} s_{p,i}}\right).$$

- So inferring this Gamma-mixture from the data gives access to the k_z, which appear as the **modes** of the functions $k_{on,i}$.

Analogy with neural network, bis



When the function $k_{on,\Theta}$ is sigmoidal, knowing the mode k_z for each cell allow to see inference as the learning of a perceptron.

The algorithm in practice

1. **Clustering step:** From a data set X, we cluster the data in m basins corresponding to m frequency modes for the promoters $k_Z = (k_z)_{z \in Z}$. We denote z_P the basin associated to a cell P.

2. **Regression step:** Find the GRN

$$\theta^* = \arg\min_{\theta} R(\theta, X) + \lambda|\theta - \theta^0|,$$

where $R(\theta, X) = \sum_{P \in X} ||k_{on}^{\theta}(P) - k_{z_P}||_2^2.$

Extension to scRNA-seq data

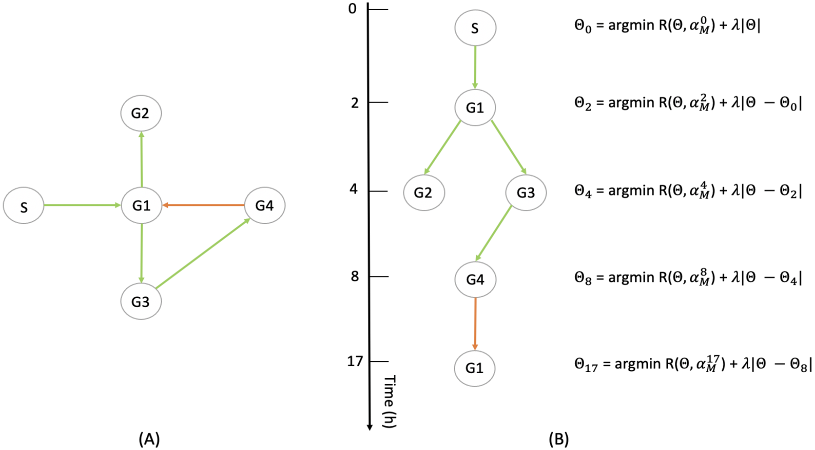
1. **Clustering step:** From a scRNA-seq data set Y, we cluster the data in m basins corresponding to m frequency modes for the promoters $\alpha_Z = (\frac{k_z}{d_m})_{z \in Z}$.

2. **Regression step:** Find the GRN

$$\theta^* = \arg\min_\theta R(\theta, Y) + \lambda|\theta - \theta^0|,$$

where $R(\theta, Y) = \sum_{M \in Y} ||k_{on}^\theta(\alpha_{z_M}) - \alpha_{z_M}||_2^2$.

Extension to timestamped data



$$\Theta_0 = \text{argmin } R(\Theta, \alpha_M^0) + \lambda|\Theta|$$

$$\Theta_2 = \text{argmin } R(\Theta, \alpha_M^2) + \lambda|\Theta - \Theta_0|$$

$$\Theta_4 = \text{argmin } R(\Theta, \alpha_M^4) + \lambda|\Theta - \Theta_2|$$

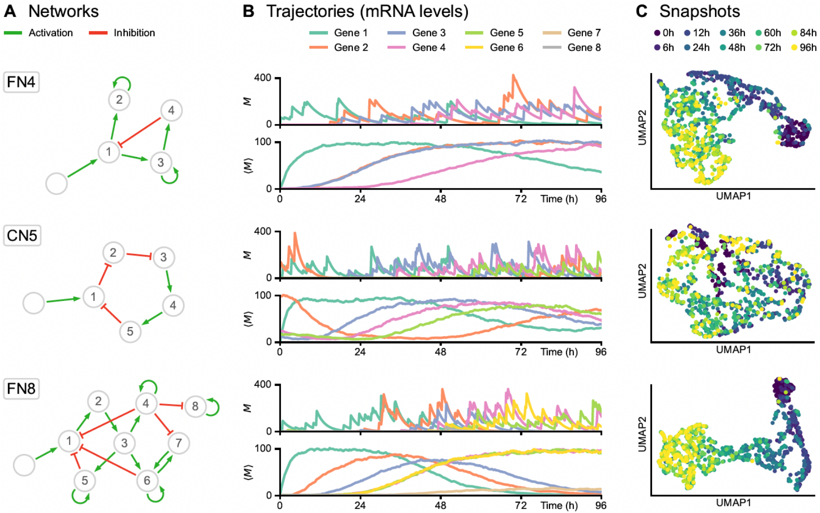
$$\Theta_8 = \text{argmin } R(\Theta, \alpha_M^8) + \lambda|\Theta - \Theta_4|$$

$$\Theta_{17} = \text{argmin } R(\Theta, \alpha_M^{17}) + \lambda|\Theta - \Theta_8|$$

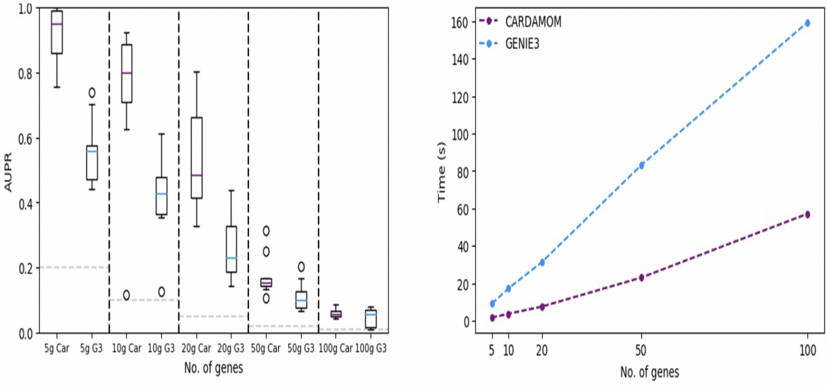
(A)

(B)

Simulated data for benchmark



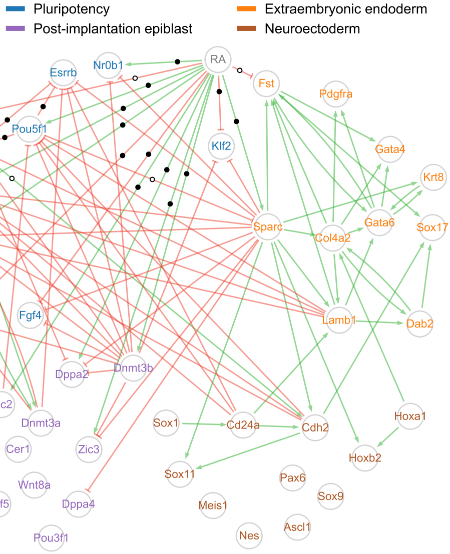
Results on tree-like networks



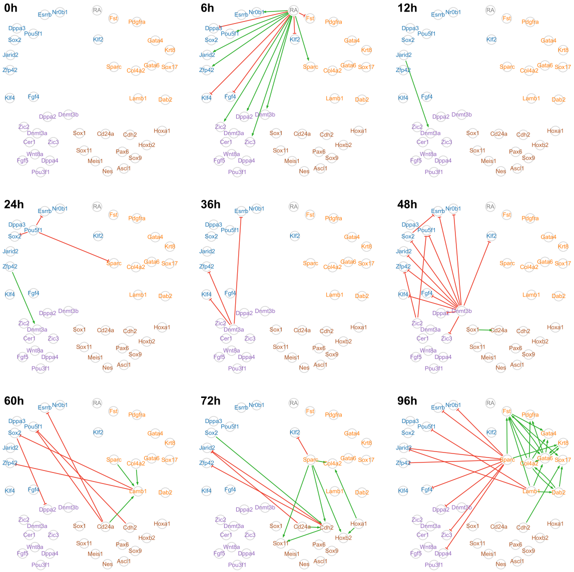
(A)

(B)

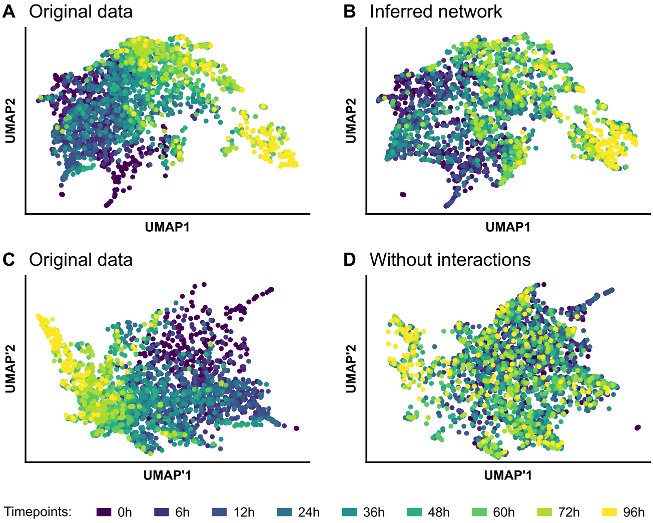
Results on real dataset (with U.Herbach, on ES cells induced by retinoic acid)



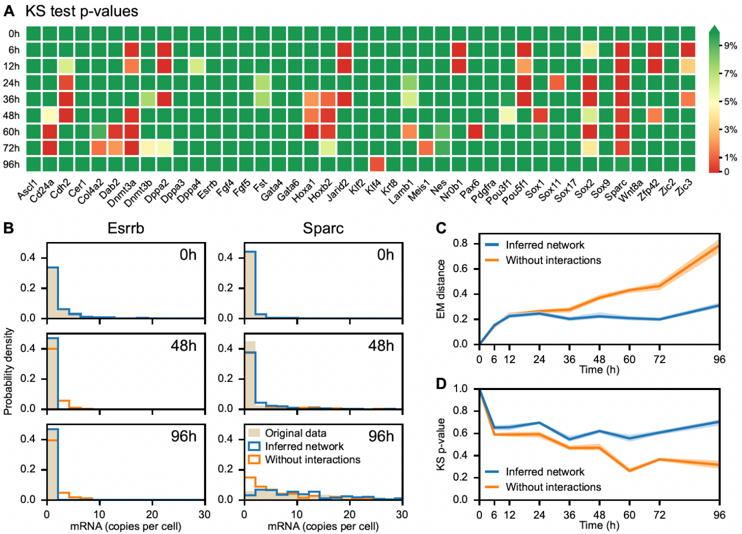
Propagation by waves of the signal



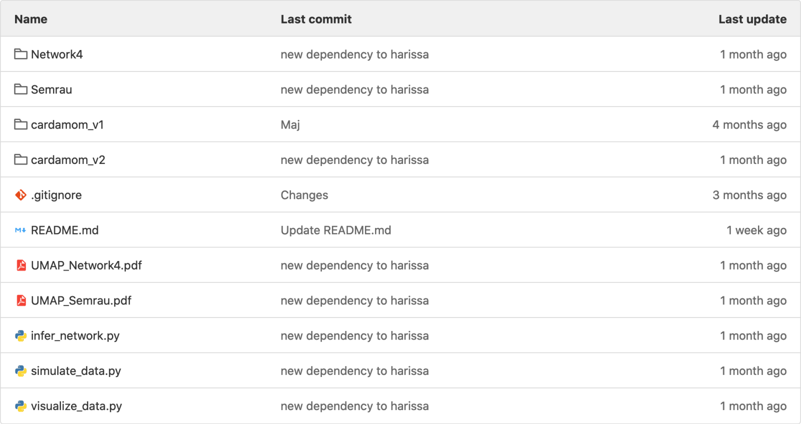
UMAP


A Original data
B Inferred network
C Original data
D Without interactions
UMAP2
UMAP1
UMAP2
UMAP1
UMAP'2
UMAP'1
UMAP'2
UMAP'1
Timepoints:
0h
6h
12h
24h
36h
48h
60h
72h
96h

Marginals


A KS test p-values
0h
6h
12h
24h
36h
48h
60h
72h
96h
Ascl1
Cd24a
Cdh2
Cer1
Col4a2
Dab2
Dnmt3a
Dnmt3b
Dppa2
Dppa3
Dppa4
Esrrb
Fgf4
Fgf5
Fst
Gata4
Gata6
Hoxa1
Hoxb2
Jarid2
Klf2
Klf4
Krt8
Lamb1
Meis1
Nes
Nr0b1
Pax6
Pdgfra
Pou3f1
Pou5f1
Sox1
Sox11
Sox17
Sox2
Sox9
Sparc
Wnt8a
Zfp42
Zic2
Zic3
9%
7%
5%
3%
1%
0
B
Esrrb
Sparc
0h
48h
96h
Probability density
0.0
0.2
0.4
0
10
20
30
mRNA (copies per cell)
Original data
Inferred network
Without interactions
C
EM distance
0.0
0.2
0.4
0.6
0.8
Inferred network
Without interactions
0
6
12
24
36
48
60
72
96
Time (h)
D
KS p-value
0.0
0.2
0.4
0.6
0.8
1.0
Inferred network
Without interactions
Time (h)

Package available on gitbio !



Name	Last commit	Last update
Network4	new dependency to harissa	1 month ago
Semrau	new dependency to harissa	1 month ago
cardamom_v1	Maj	4 months ago
cardamom_v2	new dependency to harissa	1 month ago
.gitignore	Changes	3 months ago
README.md	Update README.md	1 week ago
UMAP_Network4.pdf	new dependency to harissa	1 month ago
UMAP_Semrau.pdf	new dependency to harissa	1 month ago
infer_network.py	new dependency to harissa	1 month ago
simulate_data.py	new dependency to harissa	1 month ago
visualize_data.py	new dependency to harissa	1 month ago

Conclusion

An approach using metastability We used a combination of **supervised clustering and regressions** for reverse-engineering a mechanistic model. The result is an executable GRN model able to reproduce an experimental dataset while allowing biological interpretability.

- *Ventre. Reverse engineering of a mechanistic model of gene expression using metastability and temporal dynamics. In Silico Biology (2021).*
- *Ventre, Herbach et al. One model fits all: combining inference and simulation of gene regulatory networks. bioRxiv (2022).*

THANK YOU FOR YOUR ATTENTION !