DynGFN: Towards Bayesian Inference of Gene Regulatory Networks with GFlowNets

Lazar Atanackovic*, Alexander Tong*, Bo Wang, Leo J. Lee, Yoshua Bengio, Jason Hartford

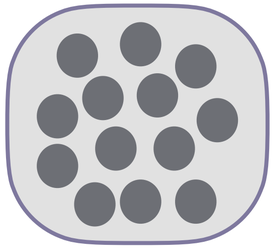
Motivation

How can we “reprogram” ***diseased*** cells back to ***healthy*** cells?

Motivation

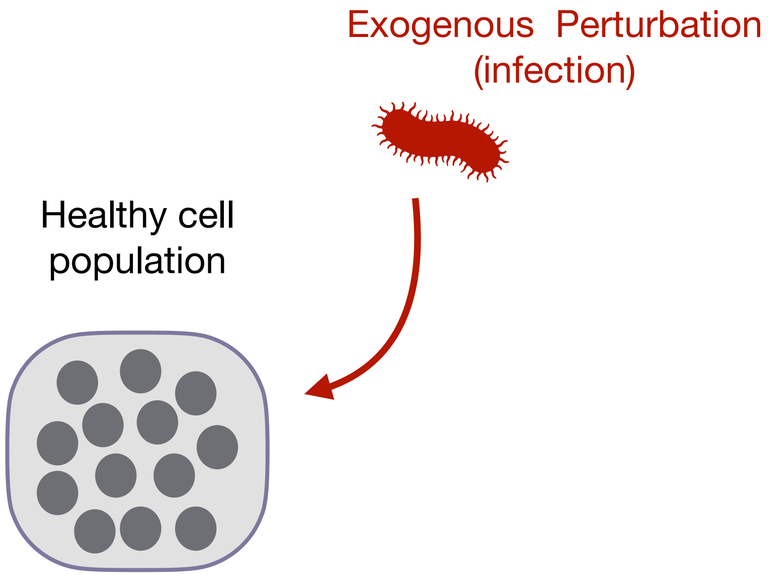
How can we “reprogram” ***diseased*** cells back to ***healthy*** cells?

Healthy cell population



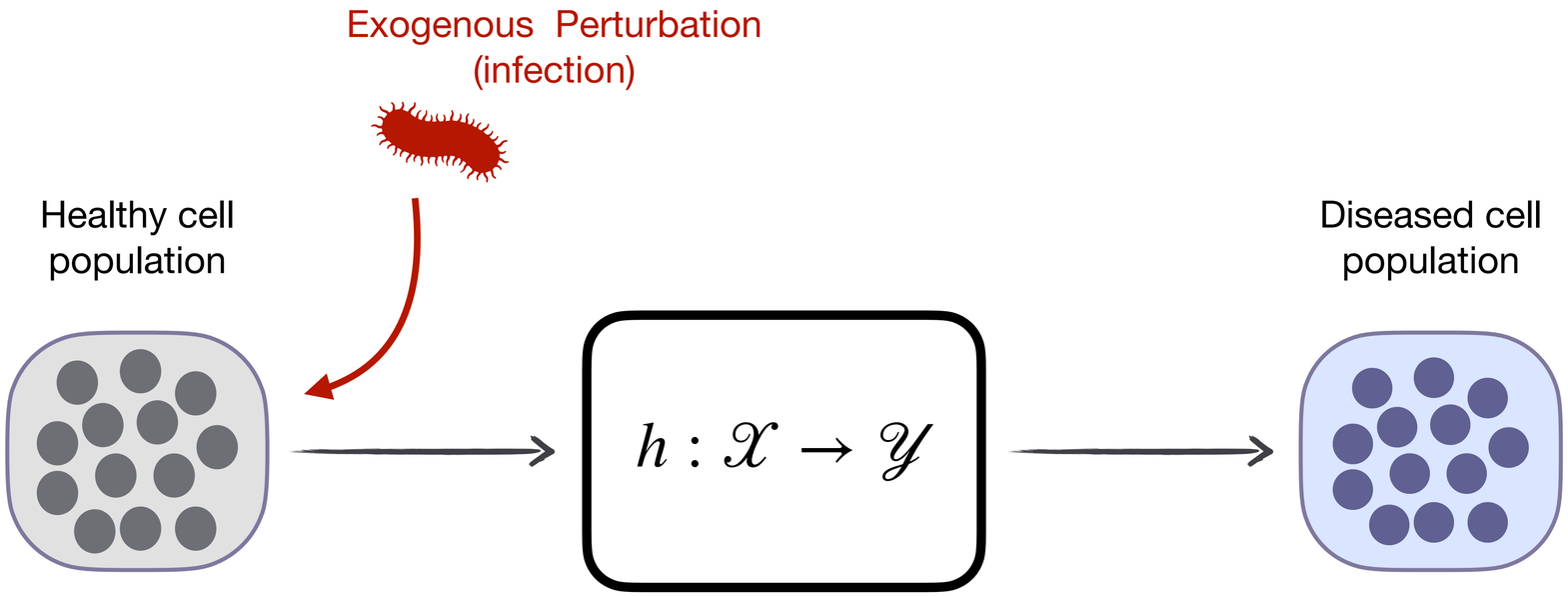
Motivation

How can we “reprogram” ***diseased*** cells back to ***healthy*** cells?



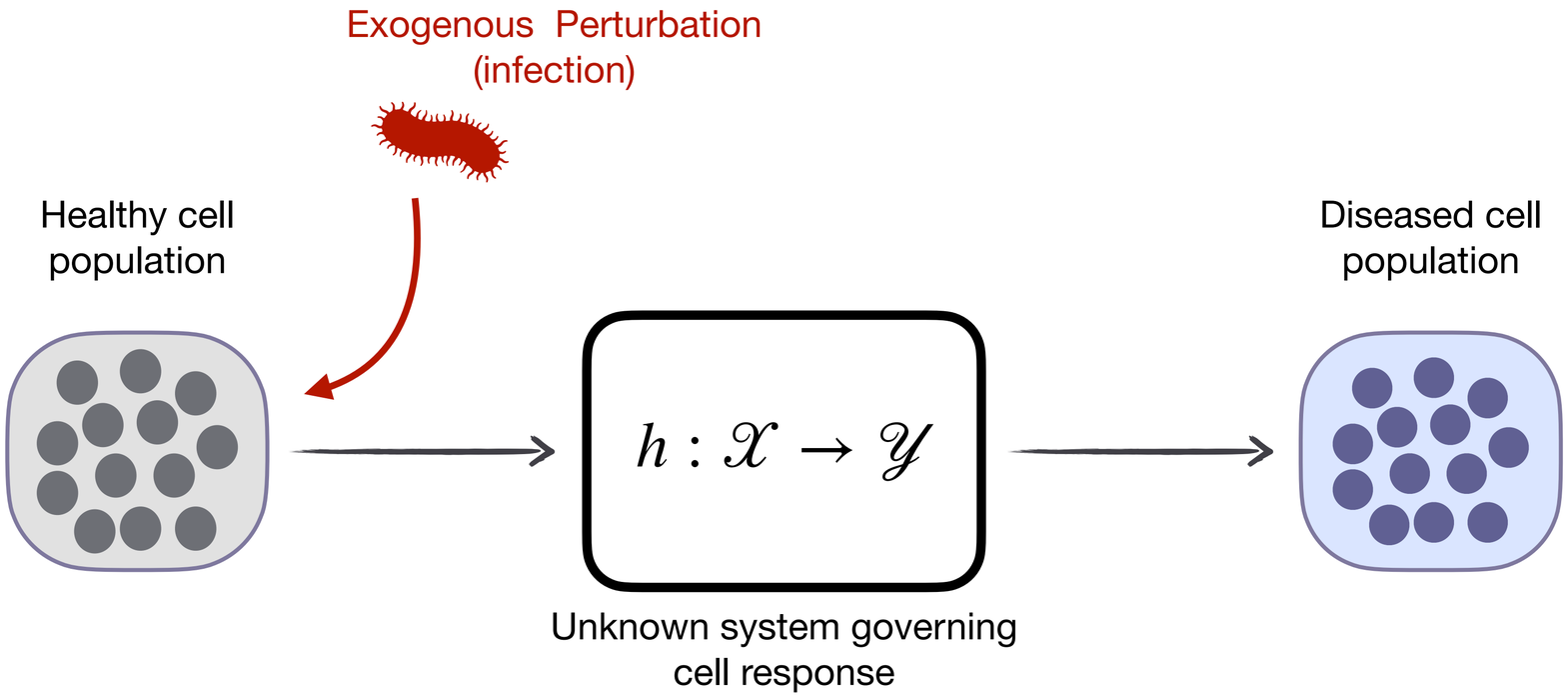
Motivation

How can we “reprogram” ***diseased*** cells back to ***healthy*** cells?



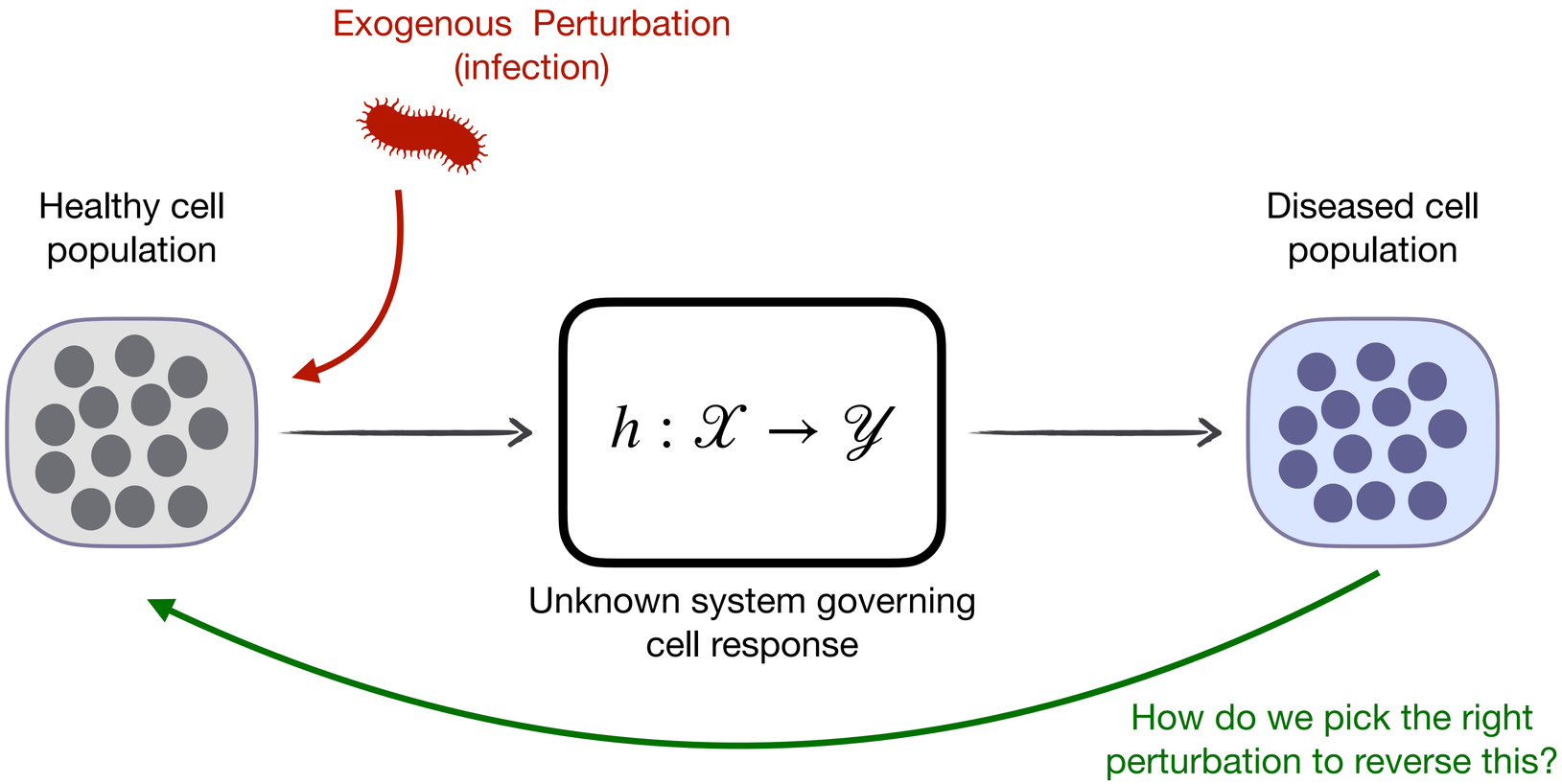
Motivation

How can we “reprogram” ***diseased*** cells back to ***healthy*** cells?



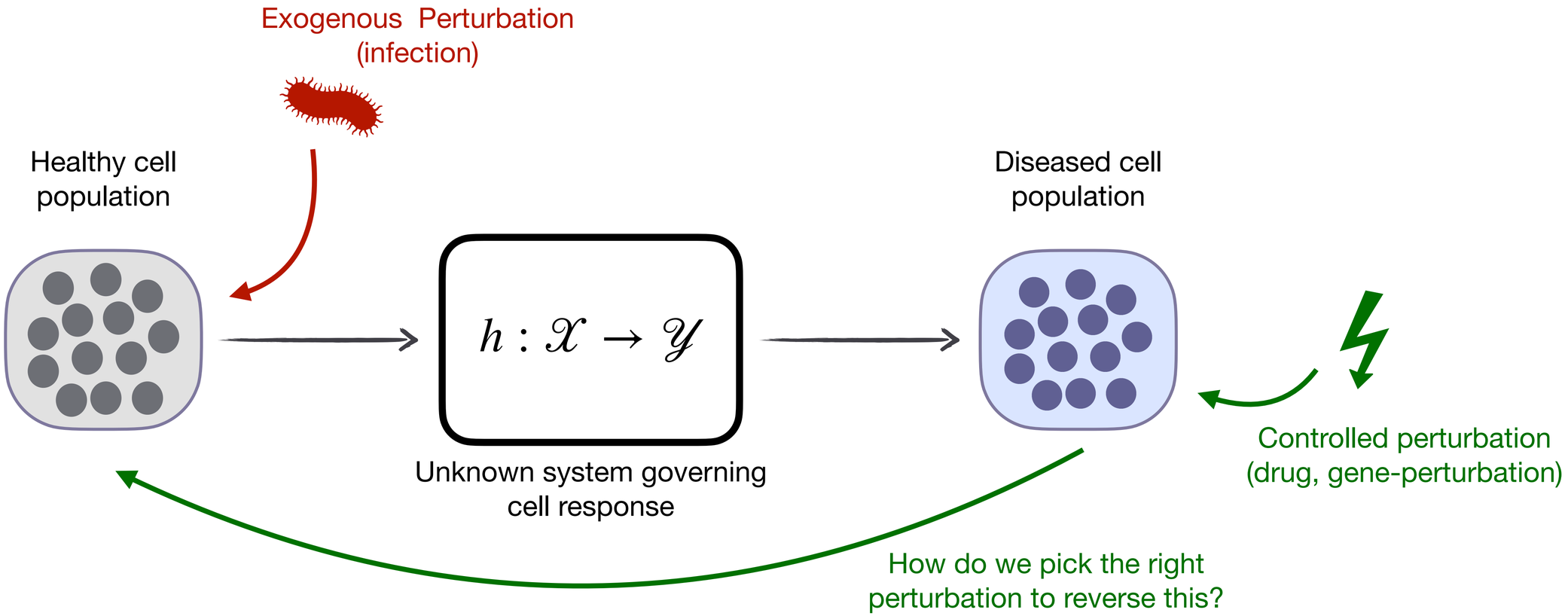
Motivation

How can we “reprogram” ***diseased*** cells back to ***healthy*** cells?



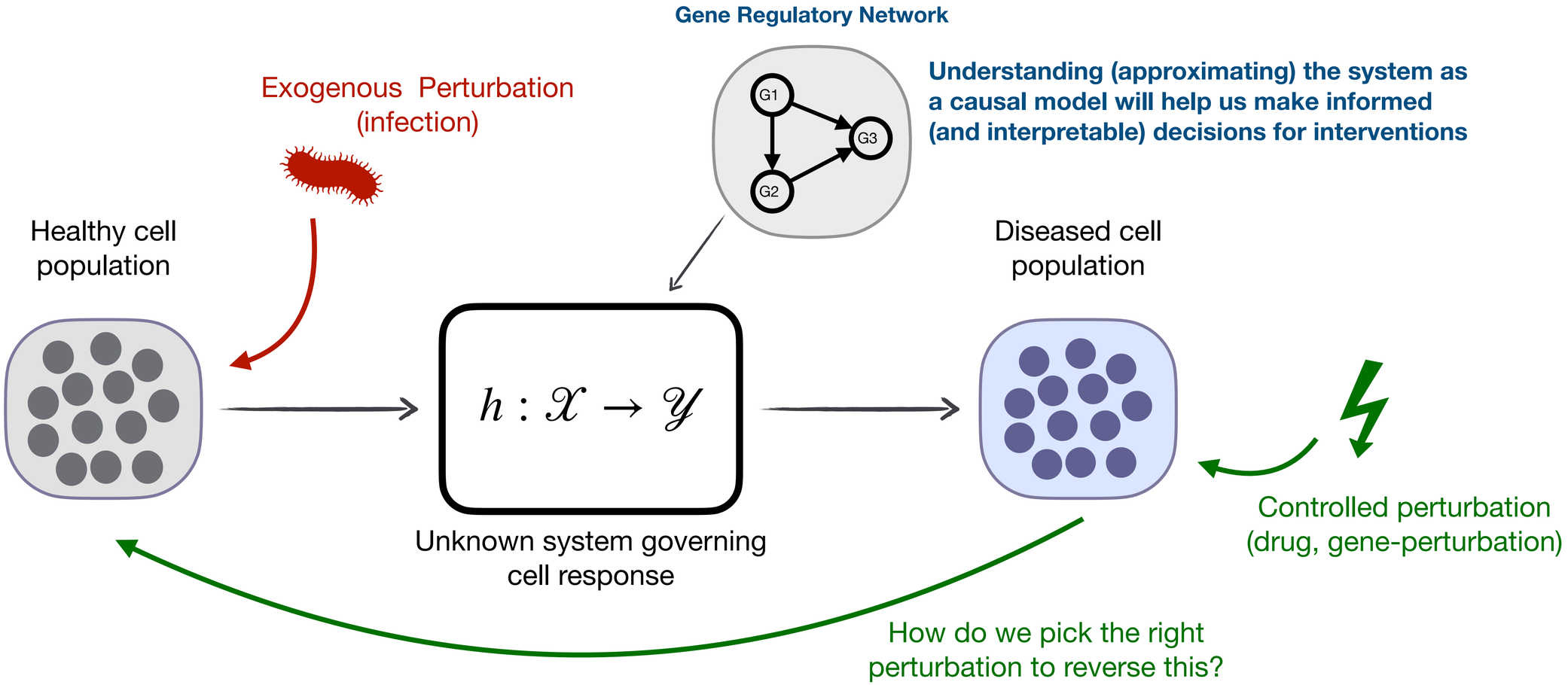
Motivation

How can we “reprogram” ***diseased*** cells back to ***healthy*** cells?



Motivation

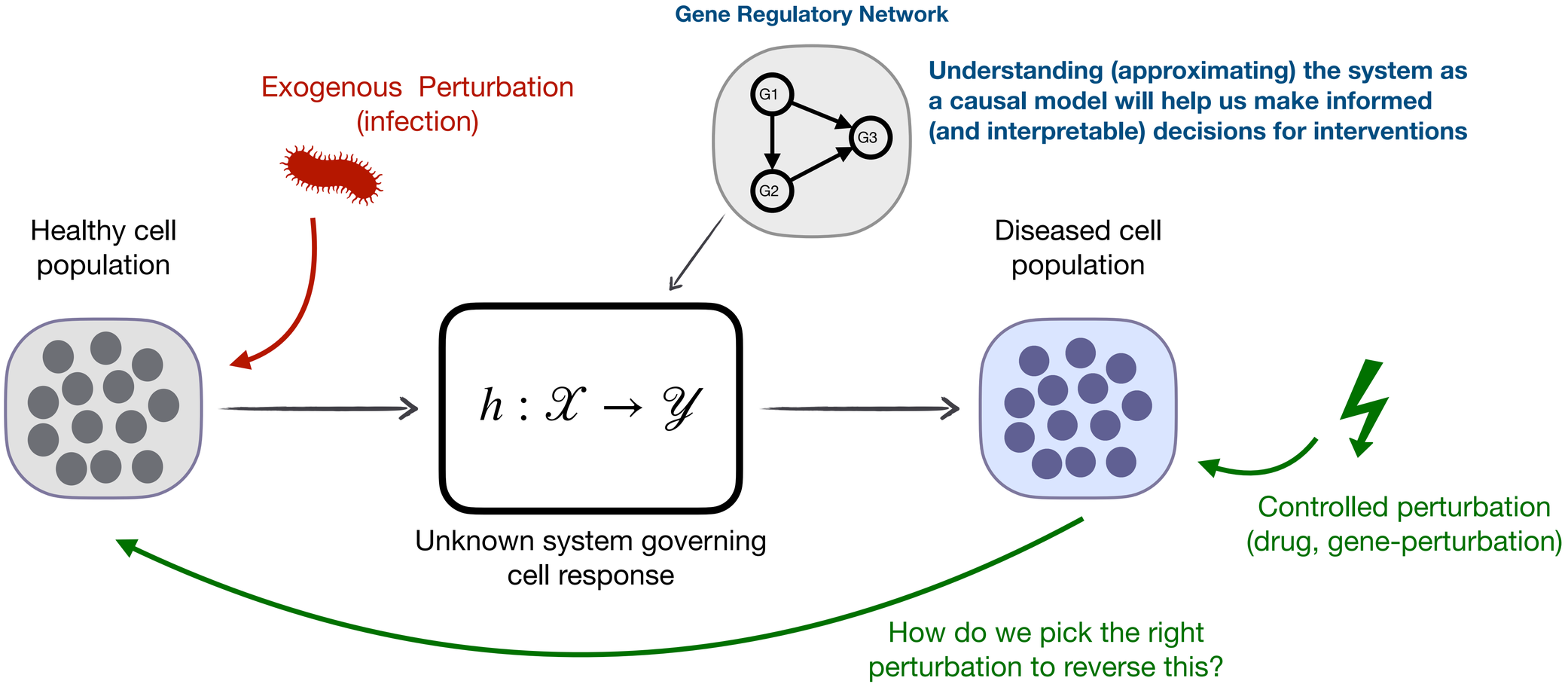
How can we “reprogram” ***diseased*** cells back to ***healthy*** cells?



Motivation

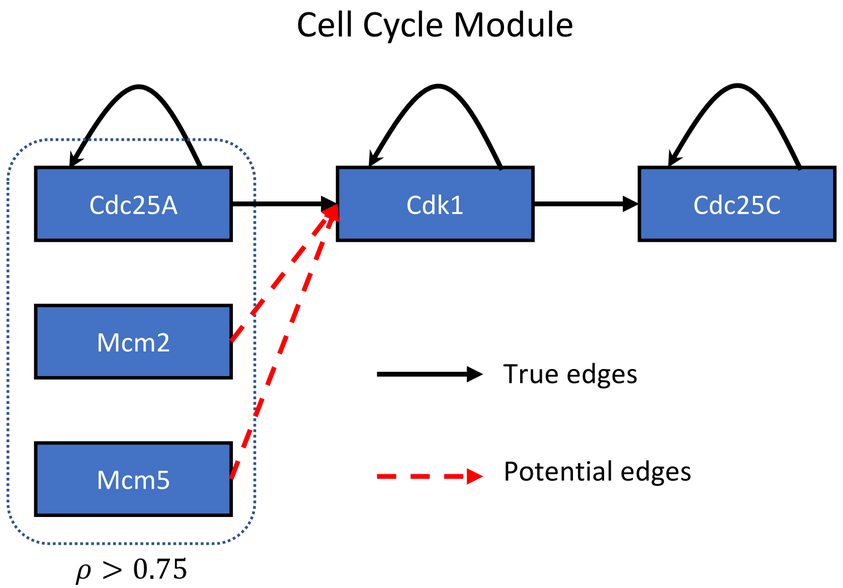
How can we “reprogram” ***diseased*** cells back to ***healthy*** cells?

How do we learn the **Gene Regulatory Network**?



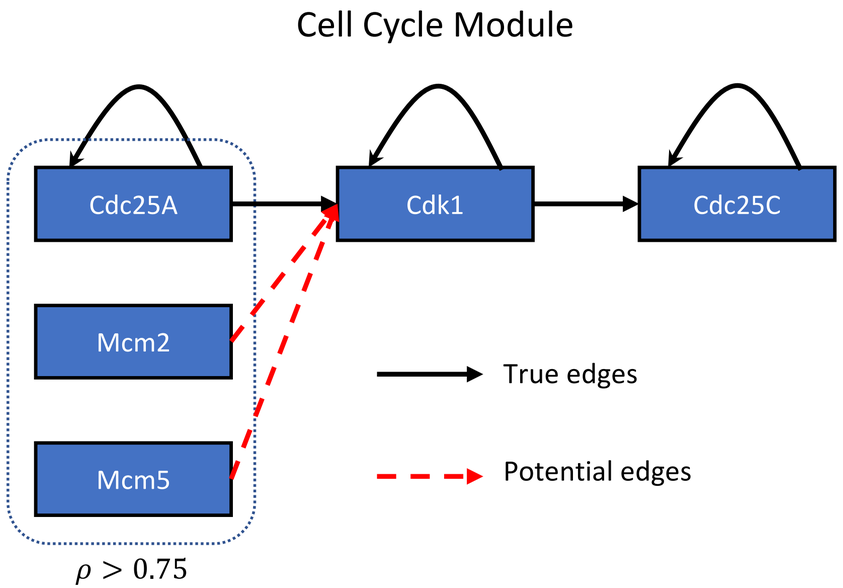
Background

- **Gene Regulatory Network (GRN)**



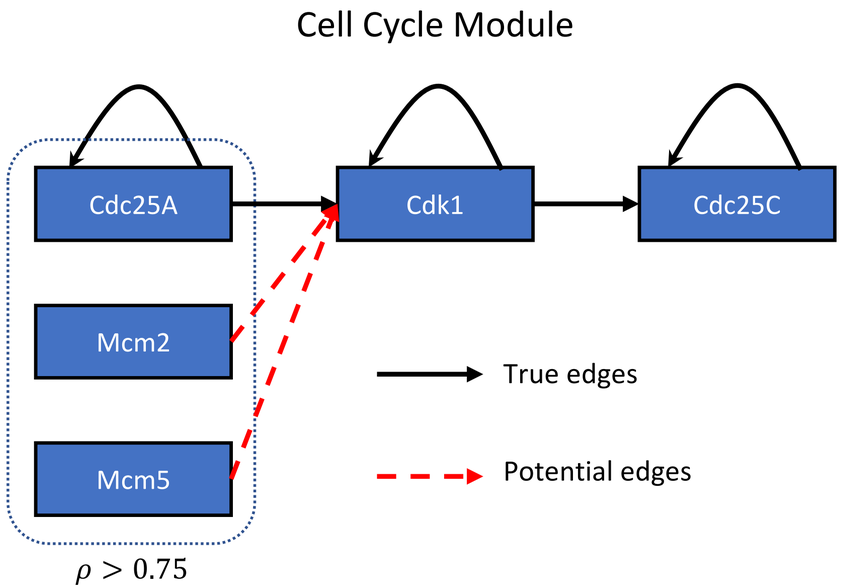
Background

- **Gene Regulatory Network (GRN)**
  - Describes interactions between genes and their products that control gene expression and cellular function



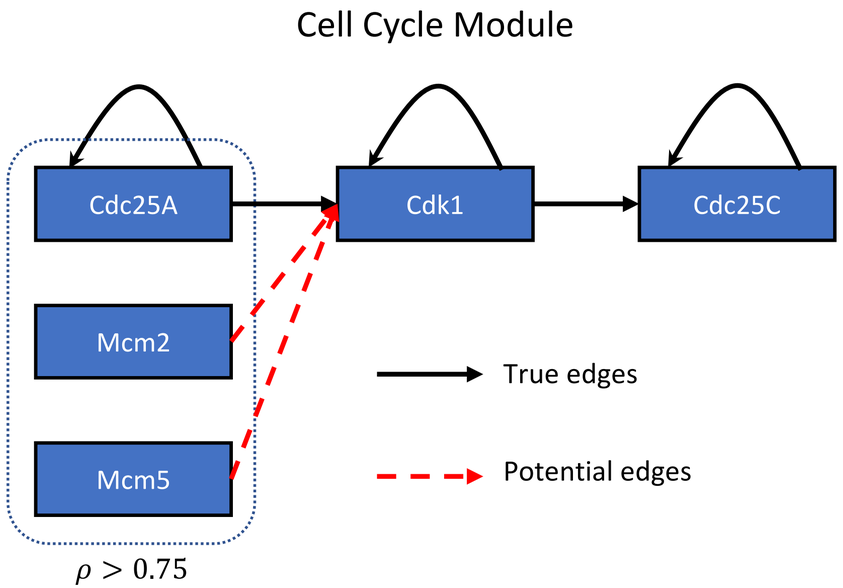
Background

- **Gene Regulatory Network (GRN)**
  - Describes interactions between genes and their products that control gene expression and cellular function
- **Gene Regulatory Network Inference**



Background

- **Gene Regulatory Network (GRN)**
  - Describes interactions between genes and their products that control gene expression and cellular function
- **Gene Regulatory Network Inference**
  - ***GRN inference*** is a ***fundamental problem*** in biology because of its applications to our understanding of cell function, drug discovery, cell development …
  - It can be cast as a ***causal structure learning*** problem

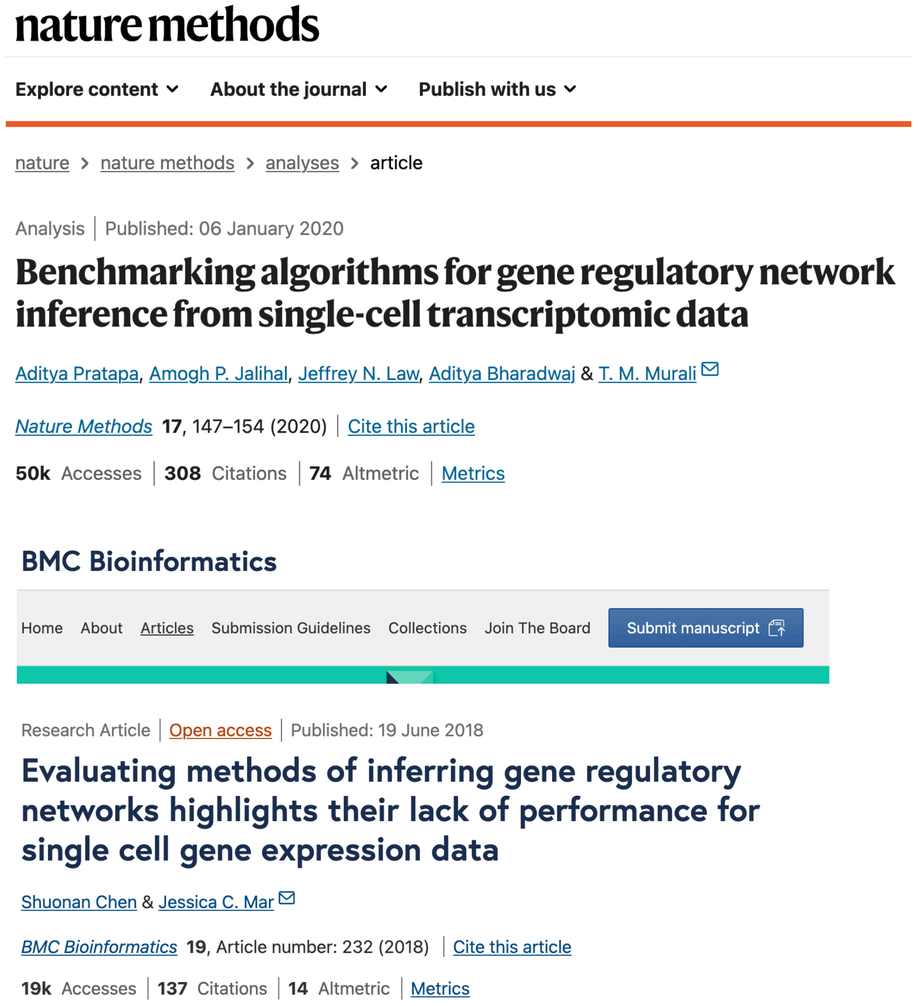


GRN inference from data is hard!

- **Causal discovery / structure learning** is generally a difficult problem.

GRN inference from data is hard!

- **Causal discovery / structure learning** is generally a difficult problem.
- Learning GRNs from data remains far from solved



nature methods

Explore content ⌄ About the journal ⌄ Publish with us ⌄

nature › nature methods › analyses › article

Analysis | Published: 06 January 2020

Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data

Aditya Pratapa, Amogh P. Jalihal, Jeffrey N. Law, Aditya Bharadwaj & T. M. Murali ✉

Nature Methods **17**, 147–154 (2020) | Cite this article

50k Accesses | **308** Citations | **74** Altmetric | Metrics

BMC Bioinformatics

Home About Articles Submission Guidelines Collections Join The Board Submit manuscript

Research Article | Open access | Published: 19 June 2018

Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data

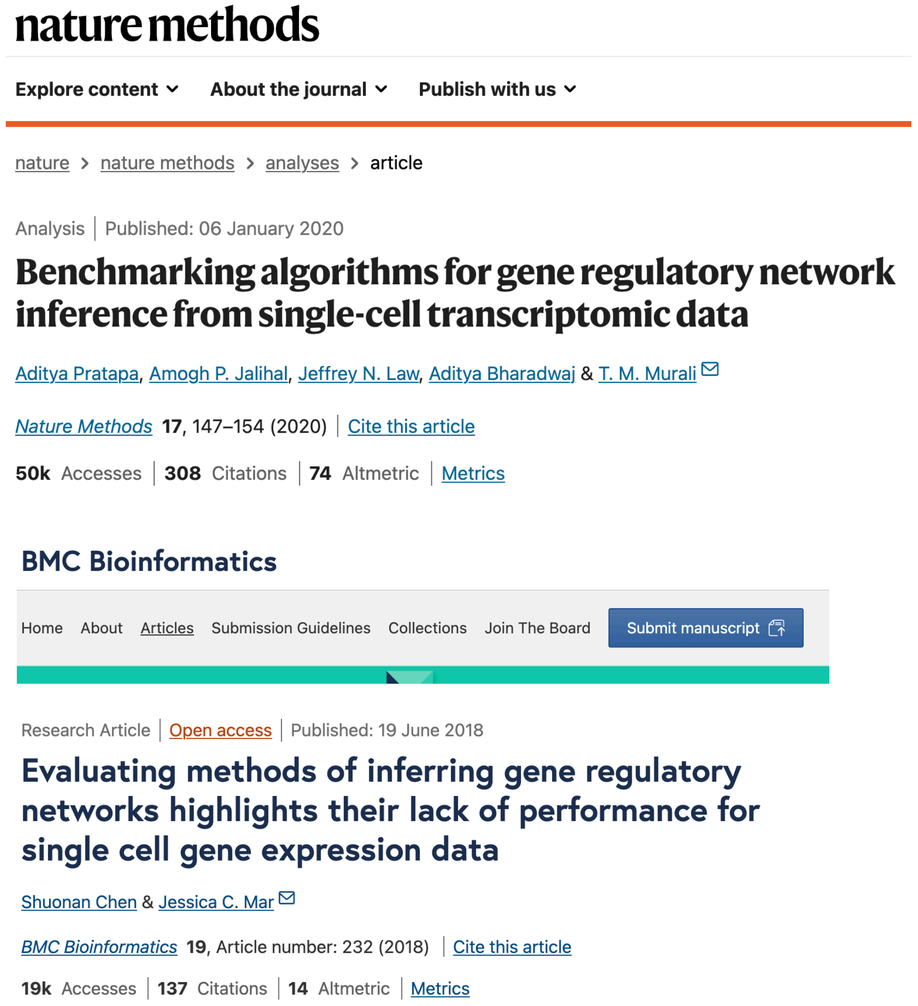
Shuonan Chen & Jessica C. Mar ✉

BMC Bioinformatics **19**, Article number: 232 (2018) | Cite this article

19k Accesses | **137** Citations | **14** Altmetric | Metrics

GRN inference from data is hard!

- **Causal discovery / structure learning** is generally a difficult problem.
- Learning GRNs from data remains far from solved
- No single method works well for every system / dataset



nature methods

Explore content ⌄ About the journal ⌄ Publish with us ⌄

nature › nature methods › analyses › article

Analysis | Published: 06 January 2020

Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data

Aditya Pratapa, Amogh P. Jalihal, Jeffrey N. Law, Aditya Bharadwaj & T. M. Murali ✉

Nature Methods **17**, 147–154 (2020) | Cite this article

50k Accesses | **308** Citations | **74** Altmetric | Metrics

BMC Bioinformatics

Home About Articles Submission Guidelines Collections Join The Board Submit manuscript

Research Article | Open access | Published: 19 June 2018

Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data

Shuonan Chen & Jessica C. Mar ✉

BMC Bioinformatics **19**, Article number: 232 (2018) | Cite this article

19k Accesses | **137** Citations | **14** Altmetric | Metrics

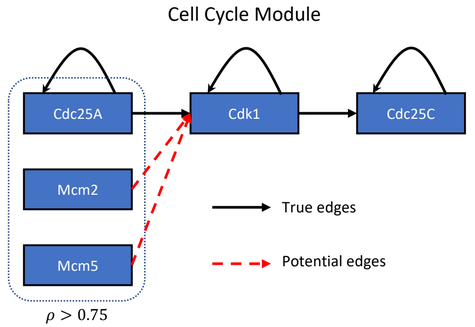
Causal Discovery for GRNs … problems

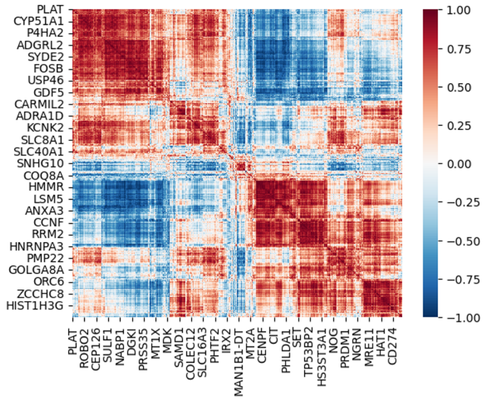
- Causal structure learning for **GRNs** come with two non-standard **challenges**:

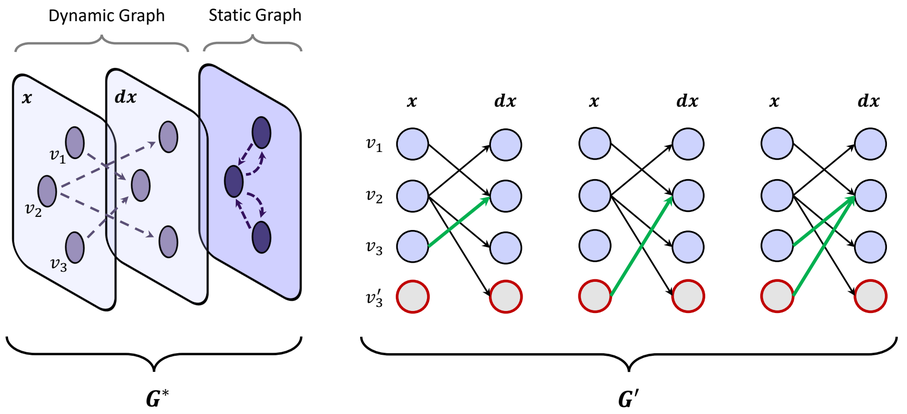
Causal Discovery for GRNs … problems

- Causal structure learning for **GRNs** come with two non-standard **challenges**:

    1. **Gene regulation contains feedback/ cycles**





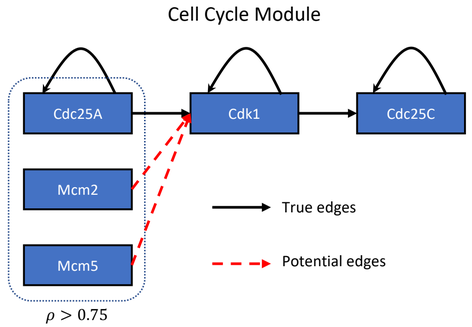


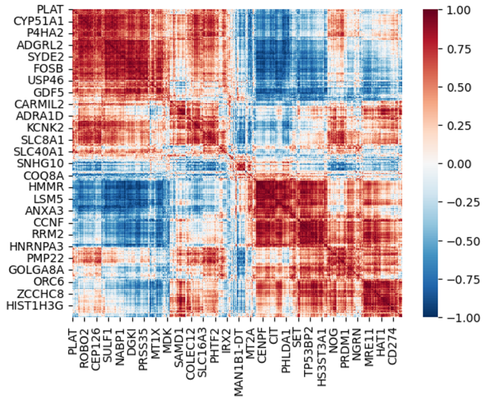
Causal Discovery for GRNs … problems

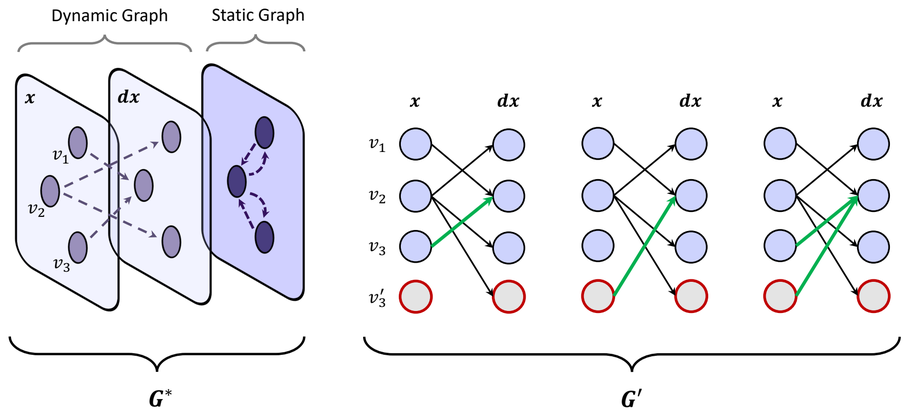
- Causal structure learning for **GRNs** come with two non-standard **challenges**:

  1. **Gene regulation contains feedback/ cycles**

  2. **Observations are limited and have measurement noise**

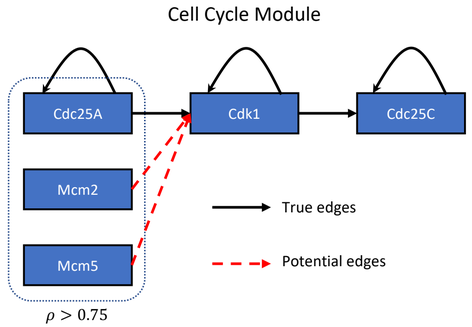


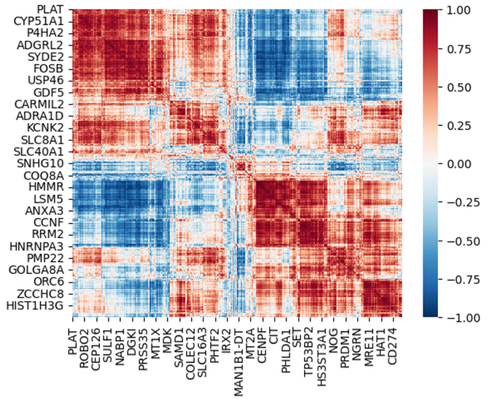


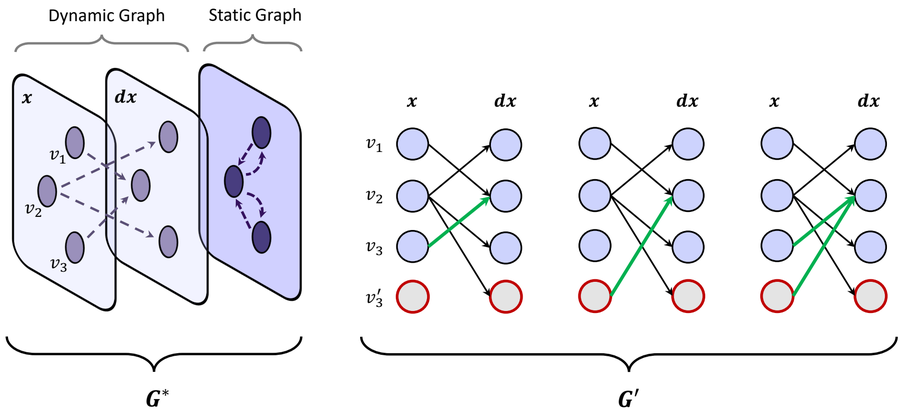


Causal Discovery for GRNs … problems

- Causal structure learning for **GRNs** come with two non-standard **challenges**:
  1. **Gene regulation contains feedback/ cycles**
  2. **Observations are limited and have measurement noise**
- This induces a multimodal distribution of graphs that explain the data

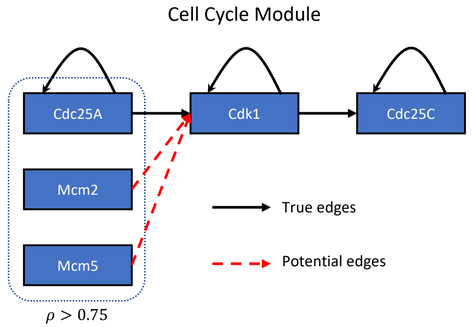


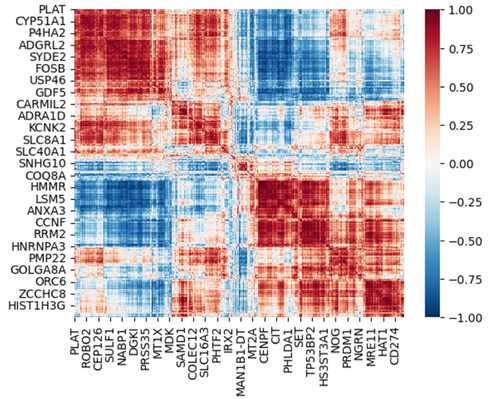


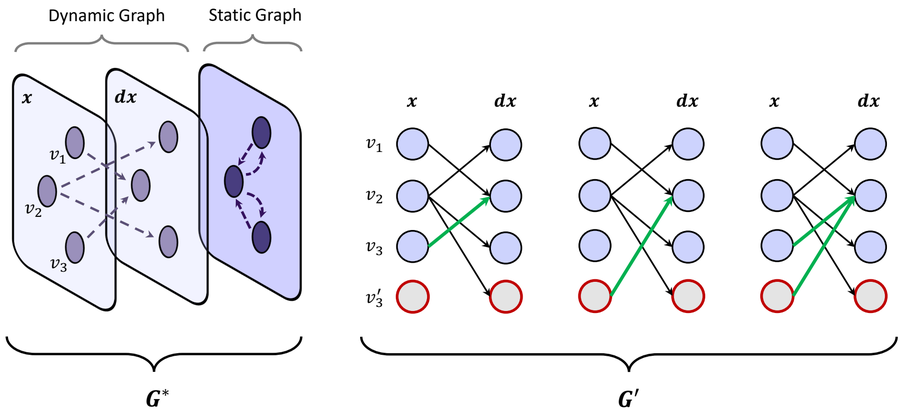


Causal Discovery for GRNs … problems

- Causal structure learning for **GRNs** come with two non-standard **challenges**:
  1. **Gene regulation contains feedback/ cycles**
  2. **Observations are limited and have measurement noise**
- This induces a multimodal distribution of graphs that explain the data
- Great … let's just model this ***uncertainty***

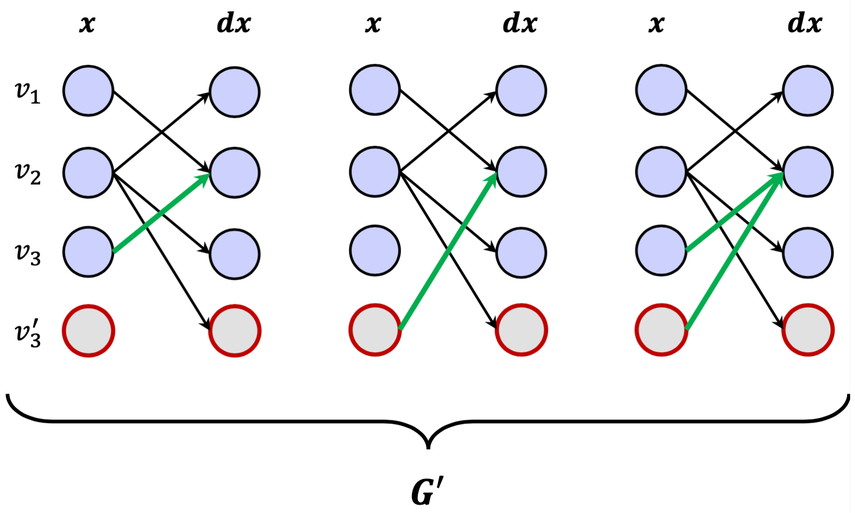






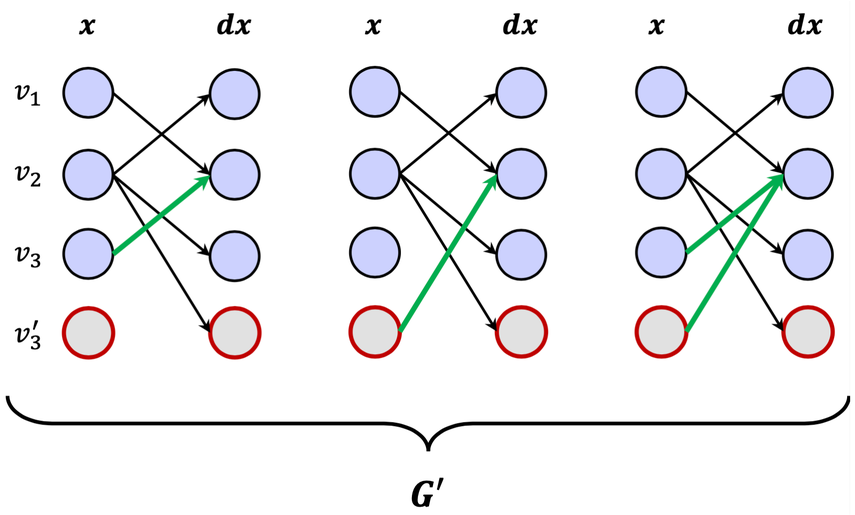
Bayesian Structure Learning

- We want to explicitly model the **uncertainty over possible structures / GRNs**



Bayesian Structure Learning

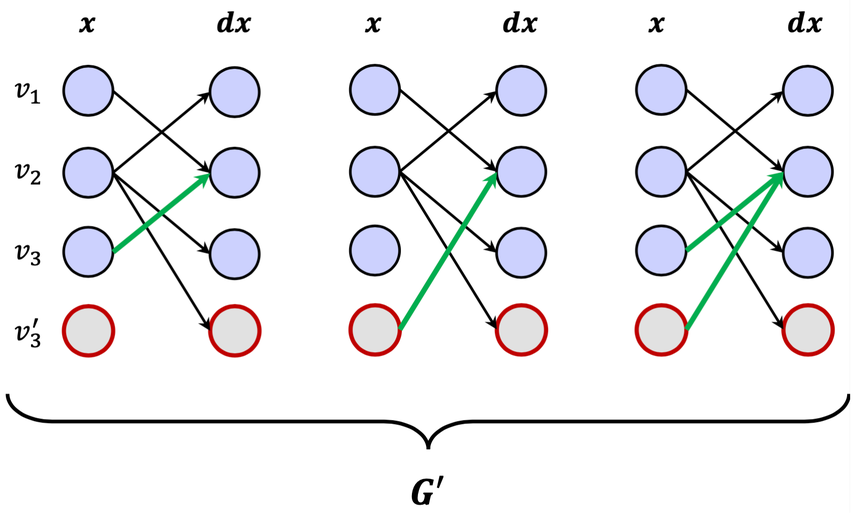
- We want to explicitly model the **uncertainty over possible structures / GRNs**
- **Objective:** learn a posterior over graphs and parameters of the structural model that best explain the data



Bayesian Structure Learning

- We want to explicitly model the **uncertainty over possible structures / GRNs**
- **Objective:** learn a posterior over graphs and parameters of the structural model that best explain the data

$$p(G, \theta \mid D) \propto p(D \mid G, \theta) p(\theta \mid G) p(G)$$

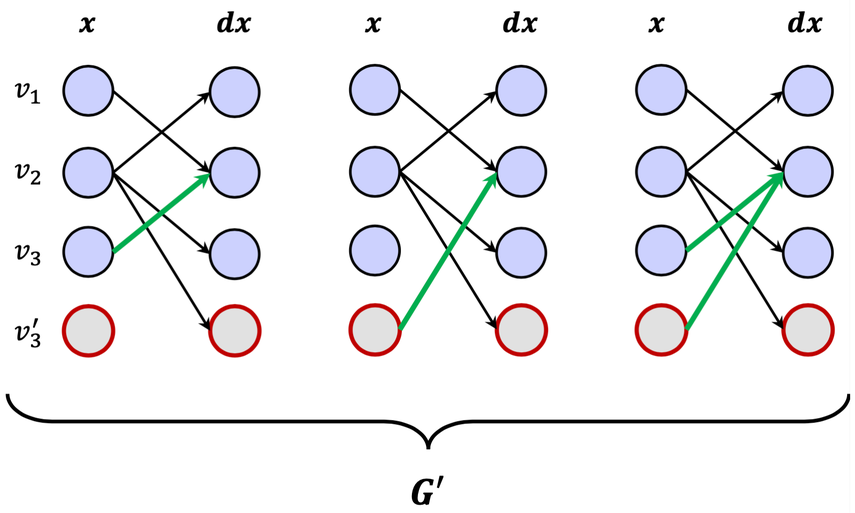


Bayesian Structure Learning

- We want to explicitly model the **uncertainty over possible structures / GRNs**
- **Objective:** learn a posterior over graphs and parameters of the structural model that best explain the data

$$p(G, \theta \mid D) \propto p(D \mid G, \theta) p(\theta \mid G) p(G)$$

- This factorization forms the basis of our inference procedure

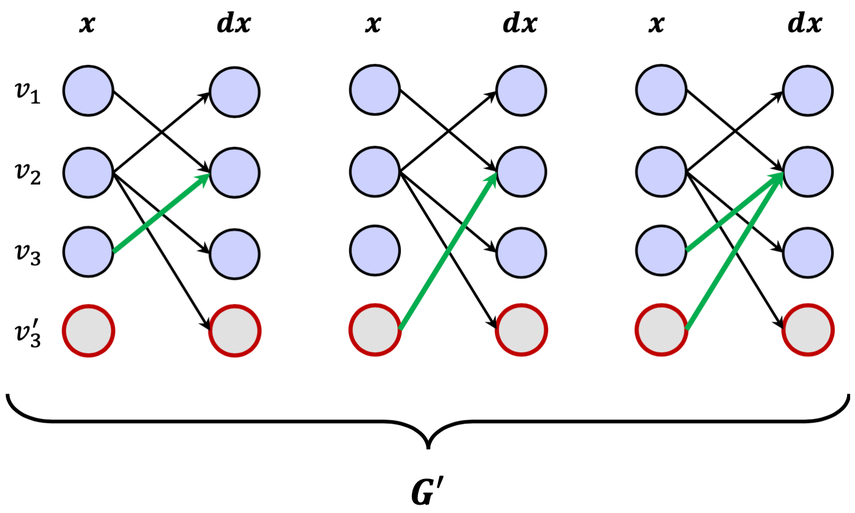


Bayesian Structure Learning

- We want to explicitly model the **uncertainty over possible structures / GRNs**
- **Objective:** learn a posterior over graphs and parameters of the structural model that best explain the data

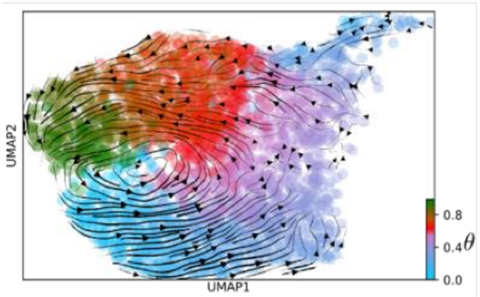
$$p(G, \theta \mid D) \propto p(D \mid G, \theta) p(\theta \mid G) p(G)$$

- This factorization forms the basis of our inference procedure
- (We can use uncertainty over edges to inform how to perturb the system, perhaps)



Structural Model

- We use the fact that we can estimate instantaneous change of gene expression (***RNA-velocity***) to acquire data of the form $D = (x, dx)$

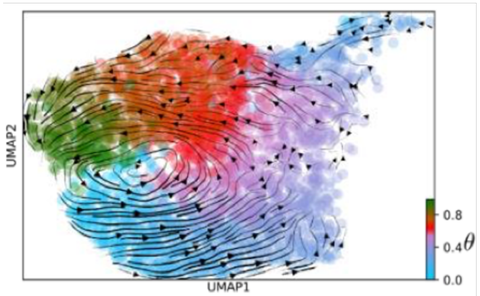


Riba, et al. *Nature Communications* (2022)

Structural Model

- We use the fact that we can estimate instantaneous change of gene expression (***RNA-velocity***) to acquire data of the form $D = (x, dx)$

- This helps us pose the problem as ***sparse*** *identification of dynamical system*



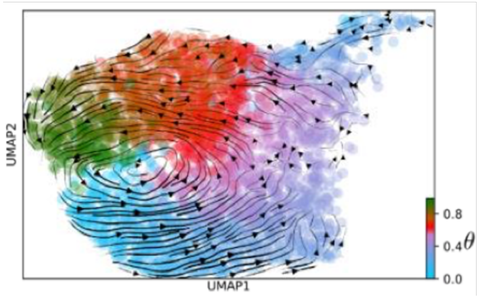
Riba, et al. *Nature Communications* (2022)

Structural Model

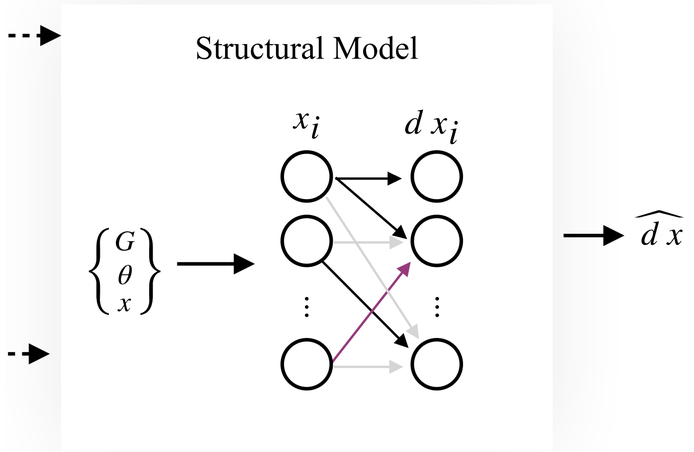
- We use the fact that we can estimate instantaneous change of gene expression (***RNA-velocity***) to acquire data of the form $D = (x, dx)$

- This helps us pose the problem as ***sparse*** *identification of dynamical system*

$$\frac{dx_i(t)}{dt} = f_i(\text{Pa}(x_i), \epsilon_i)$$



Riba, et al. *Nature Communications* (2022)



$$p(D \mid G, \theta) \propto e^{-\|dx - f_\theta(x,G)\|}$$

$$f_\theta(x, G) \rightarrow \widehat{dx}$$

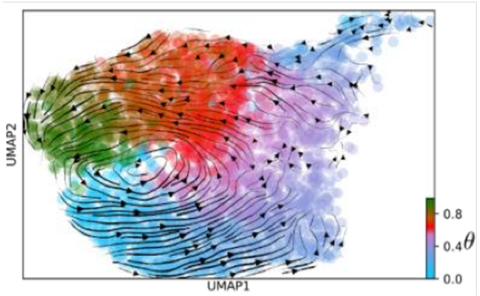
Structural Model

- We use the fact that we can estimate instantaneous change of gene expression (***RNA-velocity***) to acquire data of the form $D = (x, dx)$

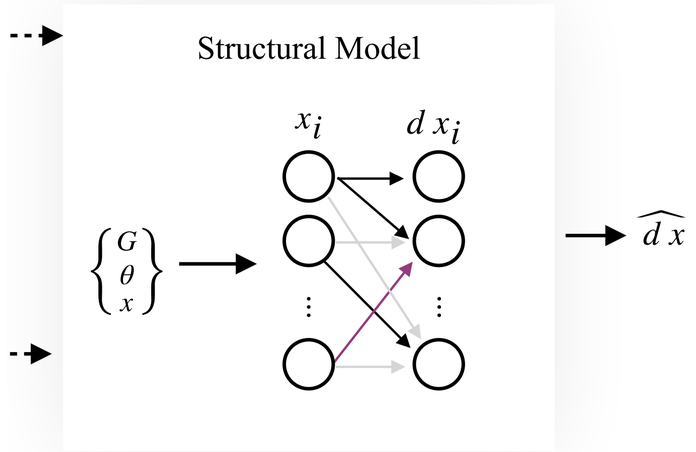
- This helps us pose the problem as ***sparse*** *identification of dynamical system*

$$\frac{dx_i(t)}{dt} = f_i(\text{Pa}(x_i), \epsilon_i)$$

- This models causal structure between genes over time (and includes ***cyclic feedback***)



Riba, et al. *Nature Communications* (2022)



$$p(D \mid G, \theta) \propto e^{-\|dx - f_\theta(x, G)\|}$$

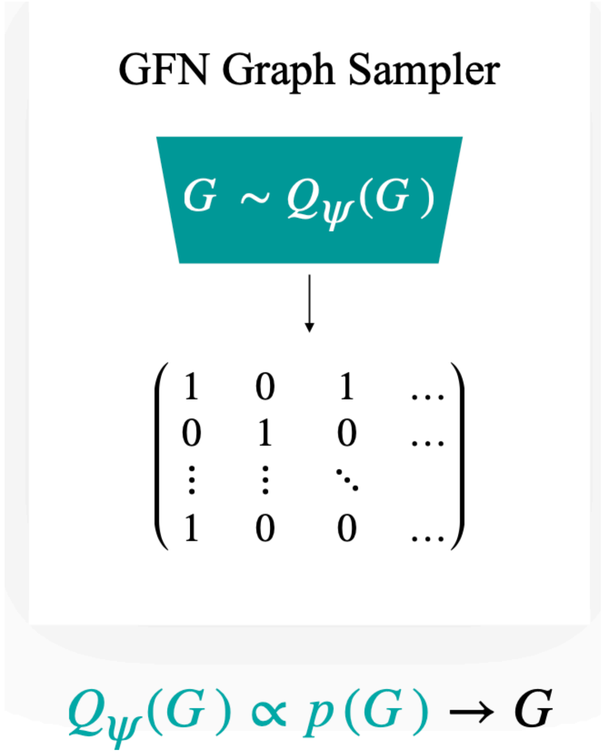
$$f_\theta(x, G) \rightarrow \widehat{dx}$$

Graph Sampler

- The graph sampler is a generative model that approximates $p(G)$ (***this is a discrete distribution***)

Graph Sampler

- The graph sampler is a generative model that approximates $p(G)$ (***this is a discrete distribution***)
- We use recent advances in **generative flow networks (GFlowNets, GFNs)** to learn a complex posterior over **discrete/graph structure**



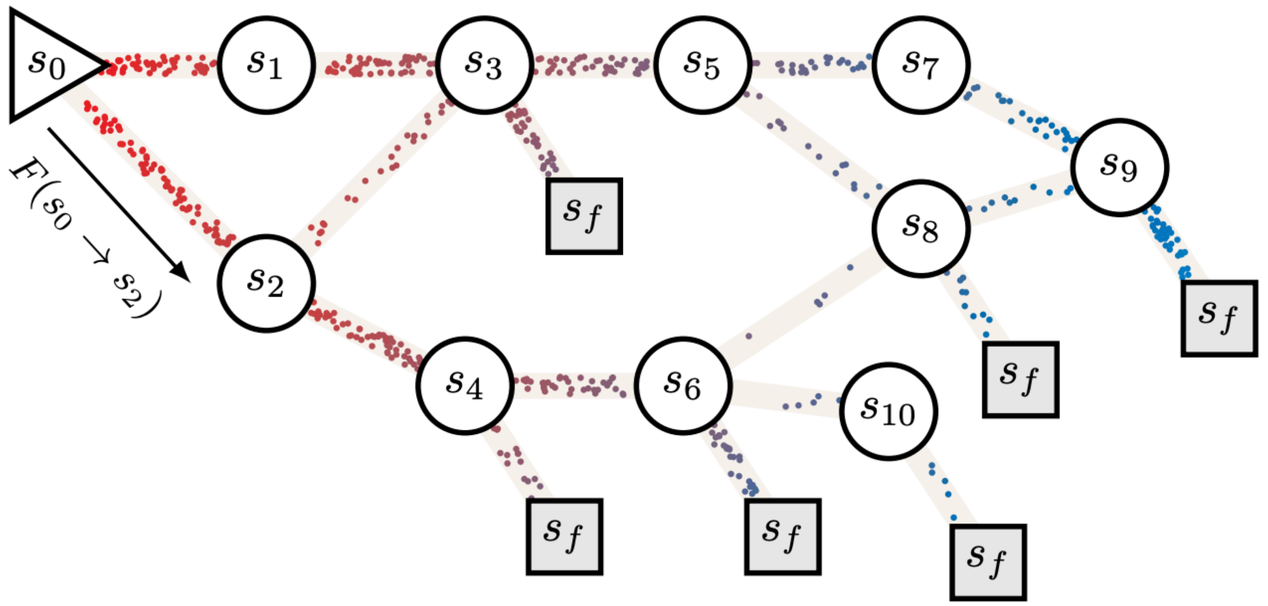
Generative Flow Networks (GFlowNets)

- Generative modelling framework for learning to sample from discrete unnormalized probability distributions

Generative Flow Networks (GFlowNets)

- Generative modelling framework for learning to sample from discrete unnormalized probability distributions
- GFlowNets learn a forward transition probability P_F that is used to sample trajectory $\tau = (s_0, \ldots, s_f)$

$$F(s_n) \prod_{i=n}^{m-1} P_F(s_{i+1}|s_i) = F(s_m) \prod_{i=n}^{m-1} P_B(s_i|s_{i+1})$$

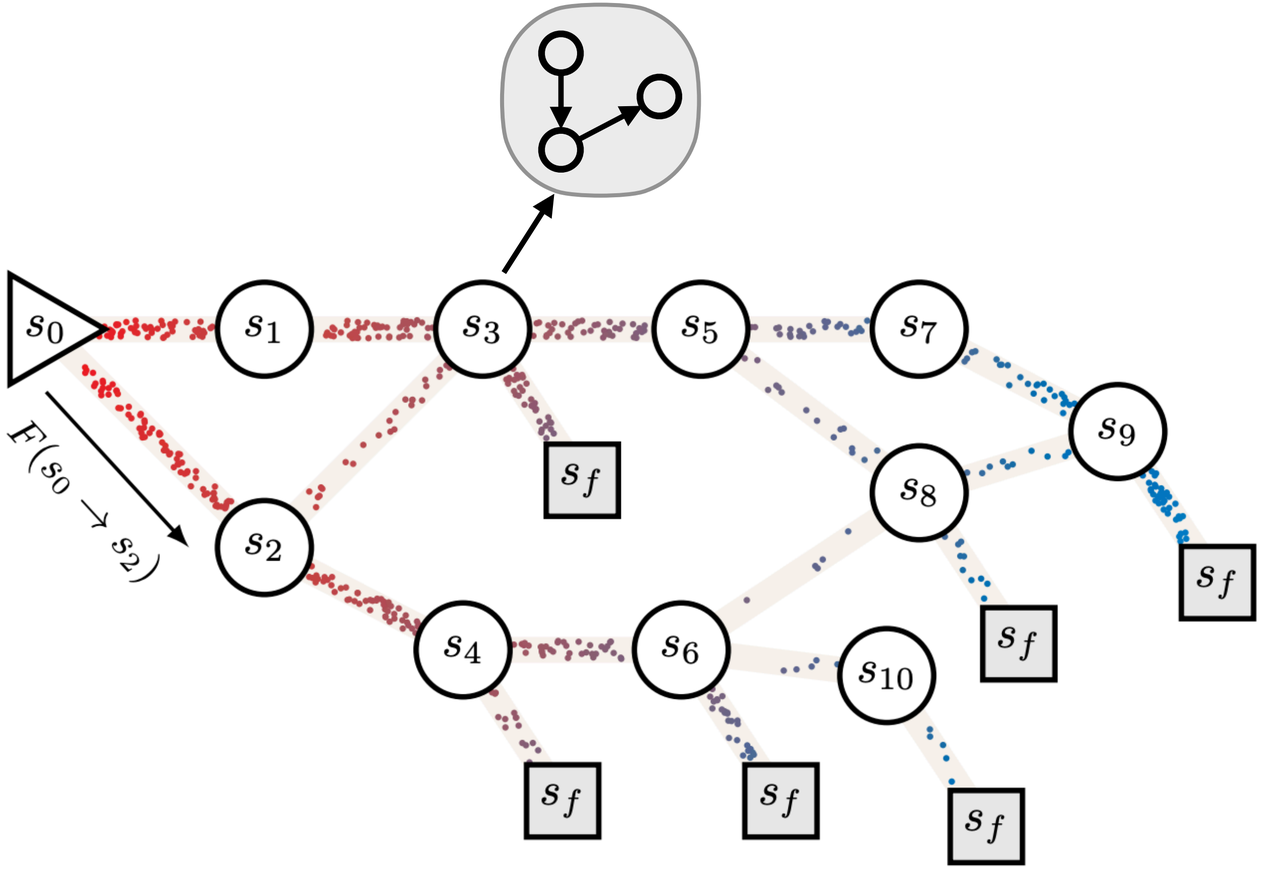


(Bengio et al. *JMLR*, 2022)

Generative Flow Networks (GFlowNets)

- Generative modelling framework for learning to sample from discrete unnormalized probability distributions
- GFlowNets learn a forward transition probability P_F that is used to sample trajectory $\tau = (s_0, \ldots, s_f)$

$$F(s_n) \prod_{i=n}^{m-1} P_F(s_{i+1}|s_i) = F(s_m) \prod_{i=n}^{m-1} P_B(s_i|s_{i+1})$$

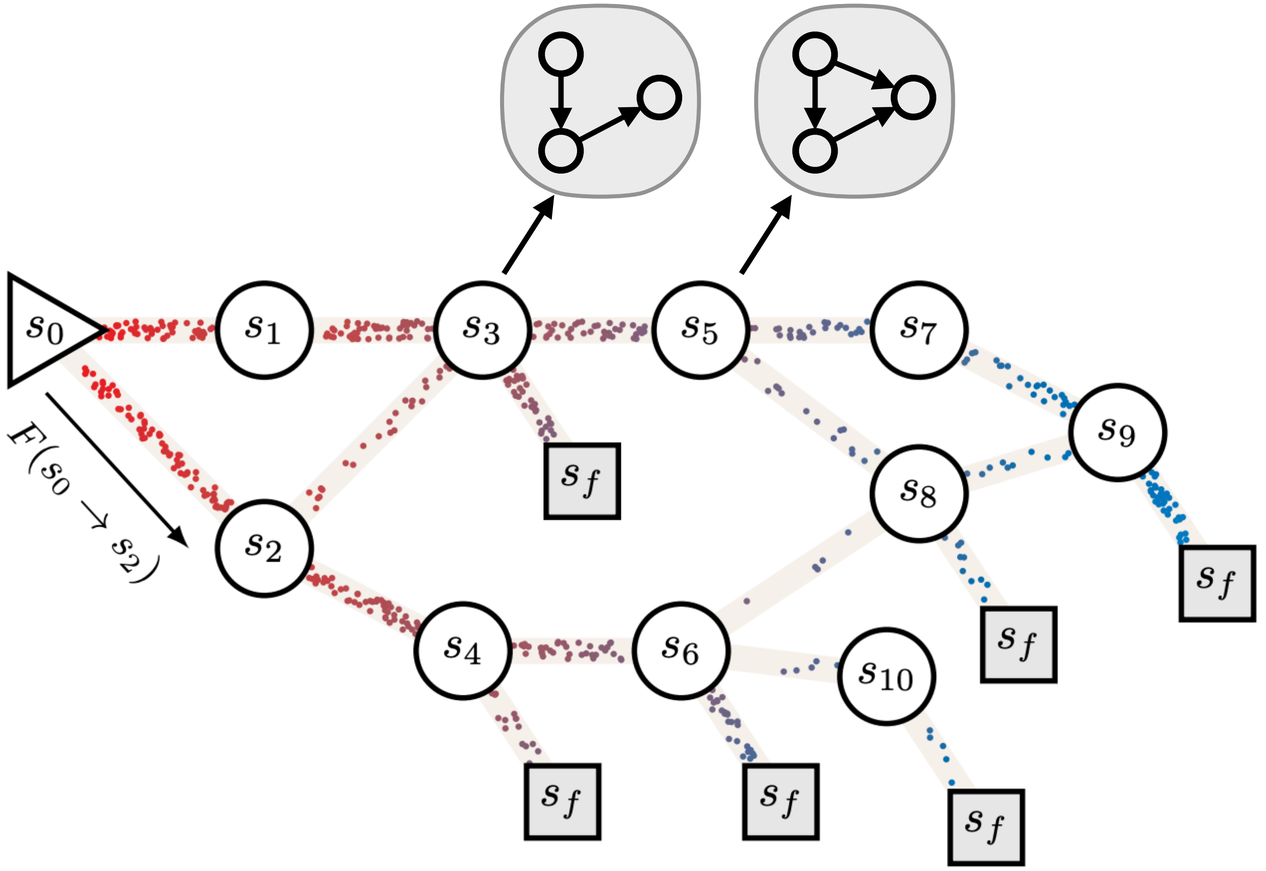


(Bengio et al. *JMLR*, 2022)

Generative Flow Networks (GFlowNets)

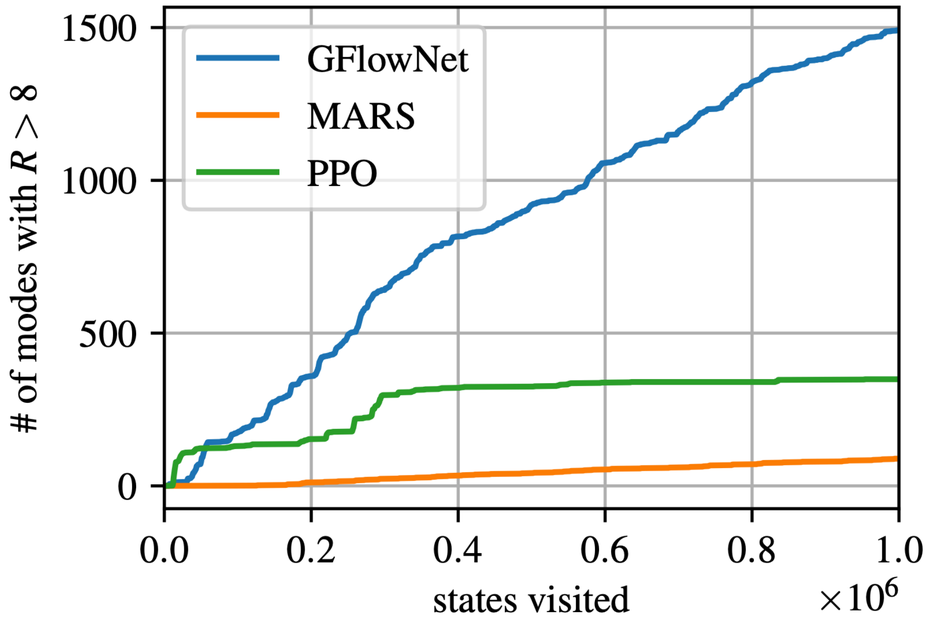
- Generative modelling framework for learning to sample from discrete unnormalized probability distributions
- GFlowNets learn a forward transition probability P_F that is used to sample trajectory $\tau = (s_0, \ldots, s_f)$

$$F(s_n) \prod_{i=n}^{m-1} P_F(s_{i+1}|s_i) = F(s_m) \prod_{i=n}^{m-1} P_B(s_i|s_{i+1})$$

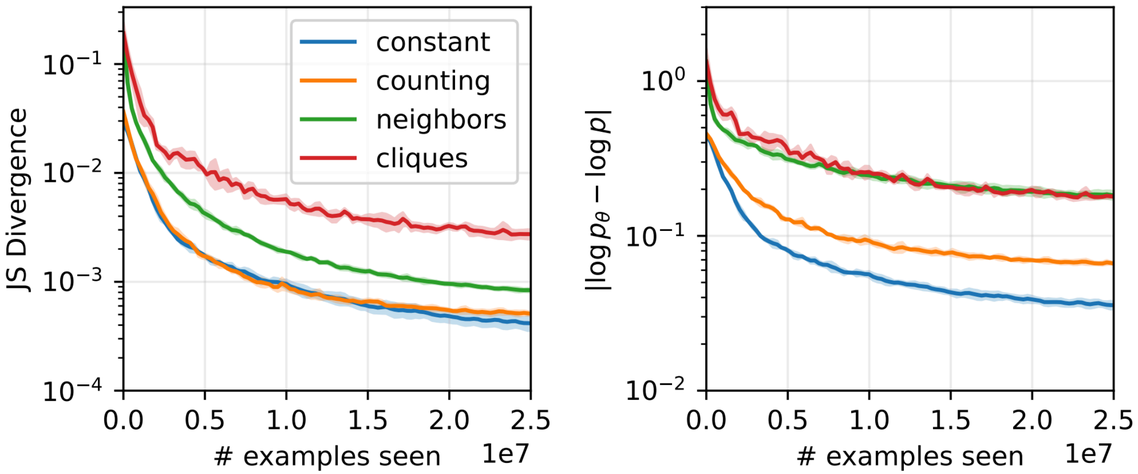


(Bengio et al. *JMLR*, 2022)

GFlowNets seem to do well in approximating multi-modal discrete distributions



(Bengio et al. NeurIPS, 2021)



(Atanackovic et al. *arxiv*, 2024)

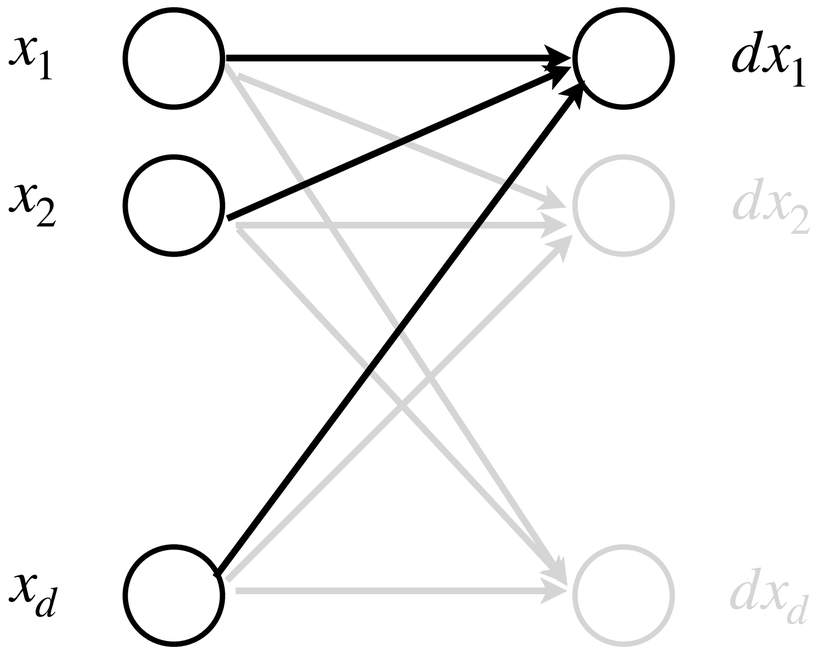
The “Per-node” GFlowNet

- Very large space of graphs G to search over
  - we can reduce this to help the model: 2^{d^2}

The “Per-node” GFlowNet

- Very large space of graphs G to search over
  — we can reduce this to help the model: 2^{d^2}
- We can exploit the independence structure of our problem to improve scalability

$$Q(G \mid D) = \prod_{i \in [1,\ldots,d]} Q_i(G[\,\cdot\,, i] \mid D)$$

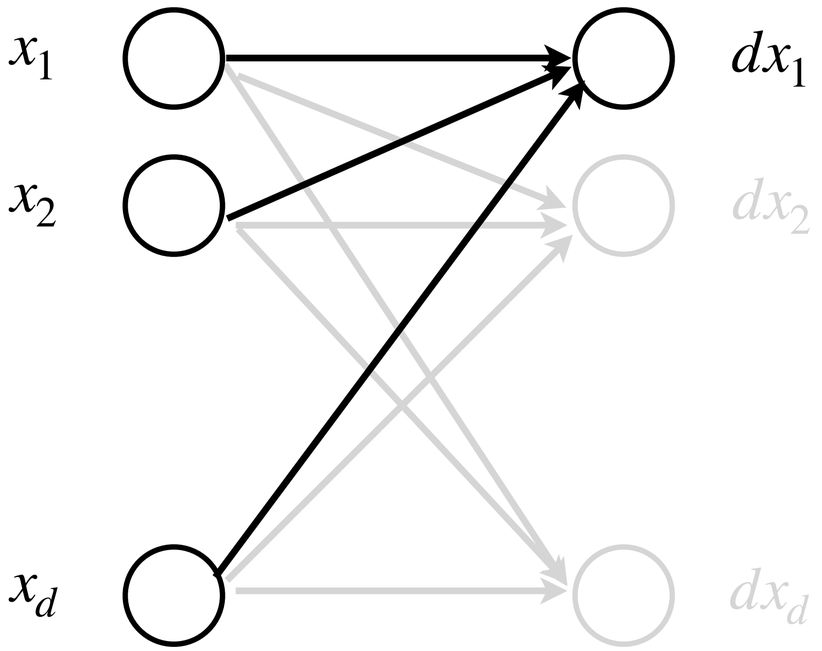


The “Per-node” GFlowNet

- Very large space of graphs G to search over
  — we can reduce this to help the model: 2^{d^2}
- We can exploit the independence structure of our problem to improve scalability

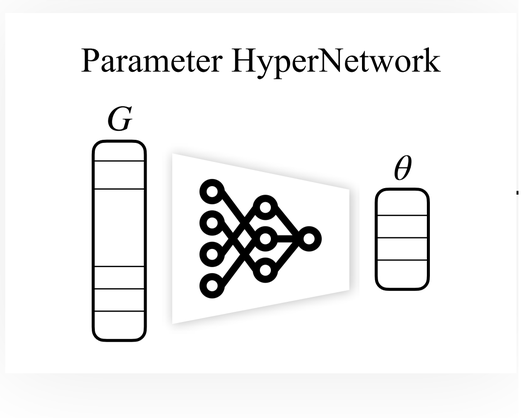
$$Q(G \mid D) = \prod_{i \in [1,\ldots,d]} Q_i(G[\,\cdot\,, i] \mid D)$$

- This way, we learn d GFNs. Thus search reduces to $d2^d$ possible graphs instead of 2^{d^2}



Parameter HyperNetwork

- Lastly, we need to learn a set of parameters θ for each respective graph G, i.e. $P(\theta \mid G)$

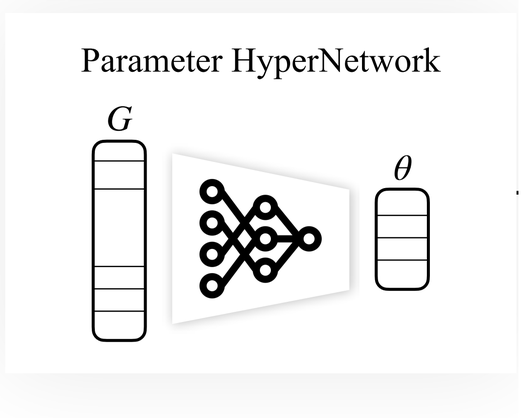


$$p(\theta \mid G) = \delta_\theta$$

$$h_\phi(G) \rightarrow \theta$$

Parameter HyperNetwork

- Lastly, we need to learn a set of parameters θ for each respective graph G, i.e. $P(\theta \mid G)$
- f_θ define the ***regulatory functions*** between variables (genes). G denotes the ***existence*** of a relationship.

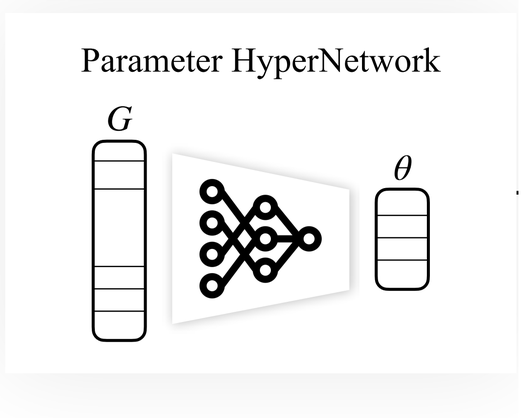


$$p(\theta \mid G) = \delta_\theta$$

$$h_\phi(G) \rightarrow \theta$$

Parameter HyperNetwork

- Lastly, we need to learn a set of parameters θ for each respective graph G, i.e. $P(\theta \mid G)$
- f_θ define the ***regulatory functions*** between variables (genes). G denotes the ***existence*** of a relationship.
- For this, we consider a **HyperNetwork** to learn the parameters of the structural model given graph structure

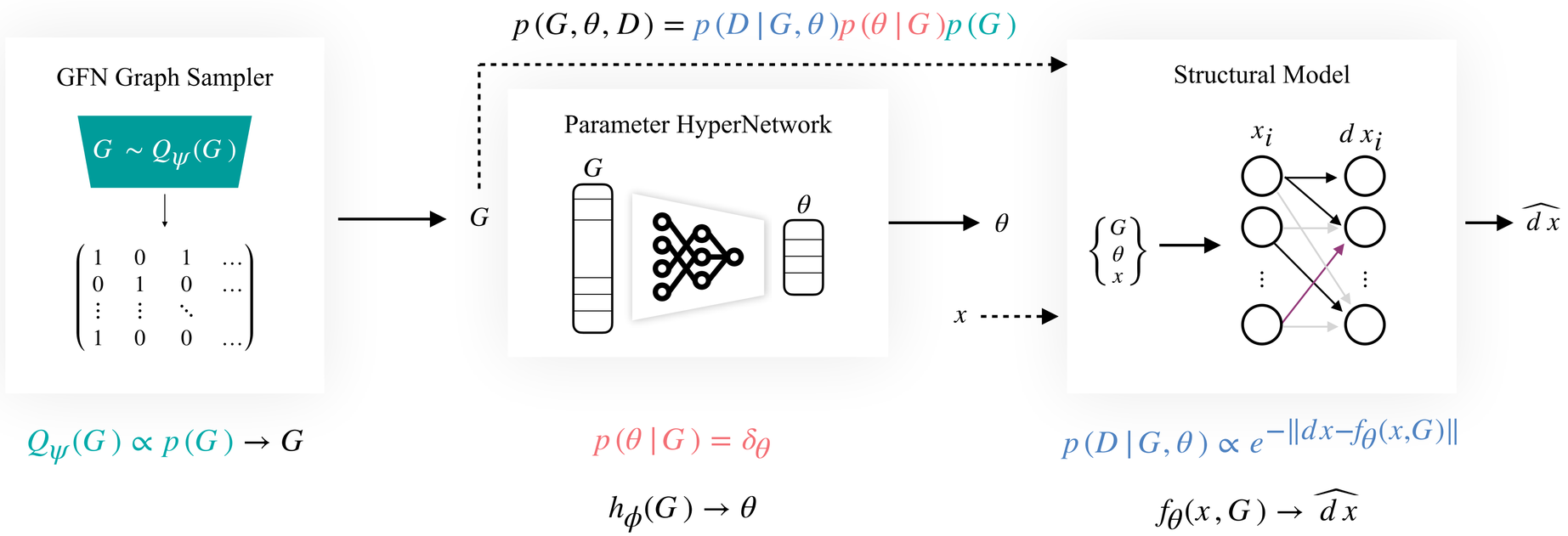


$$p(\theta \mid G) = \delta_\theta$$

$$h_\phi(G) \rightarrow \theta$$

Putting Everything Together

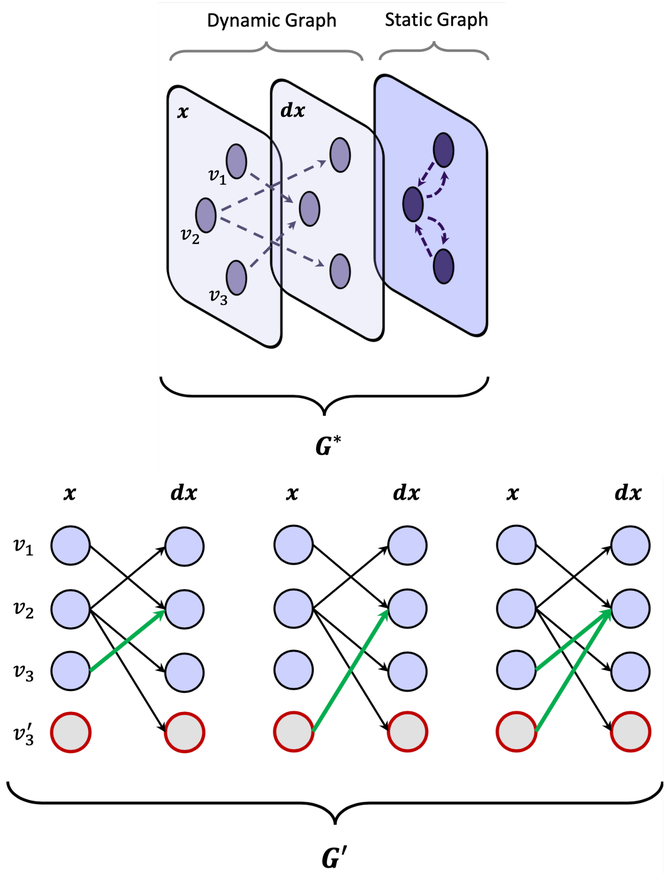
DynGFN for Bayesian Dynamic Structure Learning — a fully differentiable framework



- Note, our framework is not limited to GFlowNets (any sampler for $G \sim Q_\psi(G)$ can be used)

Synthetic Experiments

Simulating synthetic data over possible structures

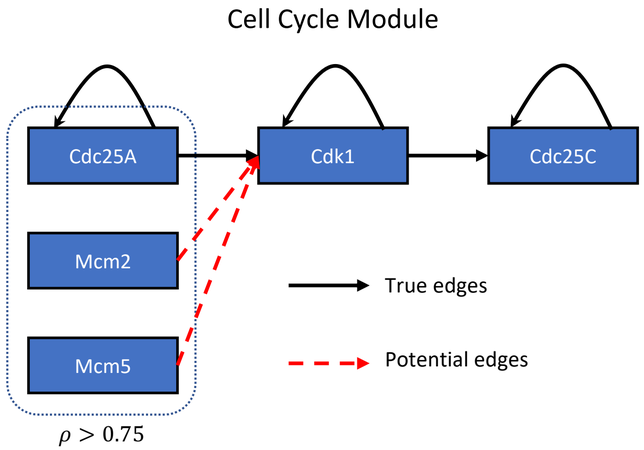


Results on synthetic 20-D linear and non-linear systems

Model	Bayes-SHD ↓	AUC ↑	KL ↓	NLL ↓
Linear System				
ℓ-DynBCD	32.0 ± 0.27	0.71 ± 0.0	1707.45 ± 9.66	—
ℓ-DynDiBS	29.2 ± 0.78	0.71 ± 0.0	6622.43 ± 171.67	—
ℓ-DynGFN	$\mathbf{22.8 \pm 1.4}$	$\mathbf{0.75 \pm 0.01}$	$\mathbf{1091.60 \pm 35.72}$	—
h-DynBCD	$\mathbf{5.5 \pm 1.1}$	0.89 ± 0.04	701.19 ± 46.99	$(9.83 \pm 0.59)\mathrm{E}-5$
h-DynDiBS	28.5 ± 4.2	0.51 ± 0.07	7934.90 ± 381.80	$(\mathbf{8.17 \pm 1.30})\mathrm{E}-6$
h-DynGFN	$\mathbf{6.7 \pm 0.0}$	$\mathbf{0.94 \pm 0.0}$	$\mathbf{350.92 \pm 30.15}$	$(8.35 \pm 0.02)\mathrm{E}-3$

Model	Bayes-SHD ↓	AUC ↑	KL ↓	NLL ↓
Non-linear System				
ℓ-DynBCD	77.5 ± 8.3	0.42 ± 0.03	3814.86 ± 354.56	—
ℓ-DynDiBS	75.7 ± 7.7	$\mathbf{0.59 \pm 0.01}$	5893.65 ± 59.66	—
ℓ-DynGFN	$\mathbf{45.7 \pm 0.6}$	0.55 ± 0.0	$\mathbf{226.25 \pm 6.58}$	—
h-DynBCD	192.9 ± 0.7	0.50 ± 0.0	9108.69 ± 51.34	$(3.83 \pm 0.32)\mathrm{E}-4$
h-DynDiBS	48.1 ± 9.0	0.53 ± 0.10	8716.64 ± 265.29	$(\mathbf{4.06 \pm 0.10})\mathrm{E}-6$
h-DynGFN	$\mathbf{32.6 \pm 0.9}$	$\mathbf{0.67 \pm 0.01}$	$\mathbf{193.28 \pm 8.53}$	$(1.47 \pm 0.11)\mathrm{E}-3$

Real Data Experiments



- We estimate dx (RNA velocity) using scVelo (Bergen et al. *Nature Biotechnology (*2020))

Results using scRNA-seq expression and velocity data for 5 genes

Model	Cellular System - RNA Velocity Bayes-SHD ↓	AUC ↑	KL ↓	NLL ↓
ℓ-DynBCD	$\mathbf{2.6 \pm 0.1}$	0.56 ± 0.01	321.95 ± 3.34	—
ℓ-DynDiBS	6.5 ± 0.4	0.47 ± 0.01	550.17 ± 16.63	—
ℓ-DynGFN	$\mathbf{3.3 \pm 0.4}$	$\mathbf{0.59 \pm 0.03}$	$\mathbf{44.98 \pm 18.60}$	—
h-DynBCD	10.1 ± 0.8	0.53 ± 0.03	587.41 ± 24.00	0.094 ± 0.003
h-DynDiBS	9.6 ± 4.2	0.51 ± 0.13	560.85 ± 83.83	$\mathbf{0.084 \pm 0.0}$
h-DynGFN	$\mathbf{5.1 \pm 1.2}$	$\mathbf{0.58 \pm 0.05}$	$\mathbf{39.82 \pm 28.05}$	0.109 ± 0.001

Conclusions and Future Work

- In low dimensions, **DynGFN** is able to better **model distributions over possible explanatory structure** compared to baselines

- **DynGFN** is able to (to some degree) **learn GRN structure from single-cell transcriptomic data**

Conclusions and Future Work

- In low dimensions, **DynGFN** is able to better **model distributions over possible explanatory structure** compared to baselines
- **DynGFN** is able to (to some degree) **learn GRN structure from single-cell transcriptomic data**
- There remains lots to still do!
  - How do we scale? (so far works only in small systems)
  - Is there a better way to approximate $p(G)$?
  - How do we incorporate interventions/perturbations?

Thanks for your Attention!

VECTOR INSTITUTE

UHN

Mila

Valence Labs
Powered by Recursion