

# Modelling single-cell dynamics with trajectories and gene regulatory networks

Robrecht Cannoodt

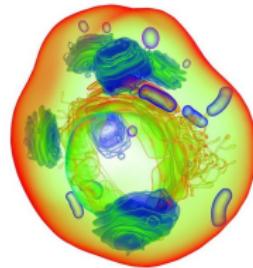


# Single-cell biology

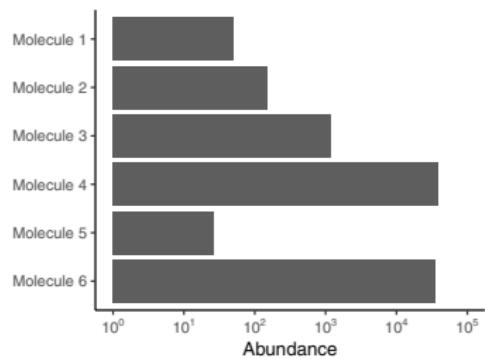
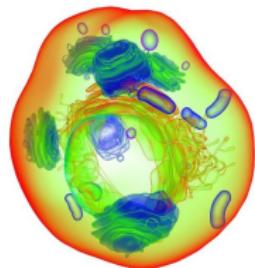
Single-cell biology is to study an organism



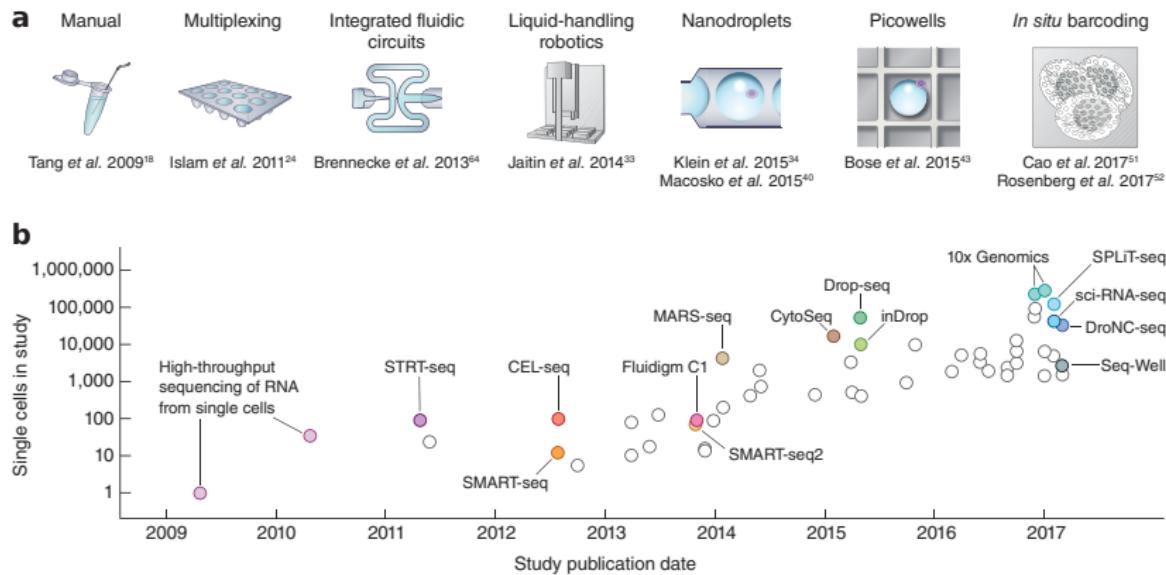
Single-cell biology is to study an organism  
by studying the behaviour of its cells



Single-cell biology is to study an organism  
by studying the behaviour of its cells



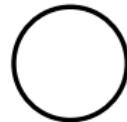
# Technological advances allow high-throughput single-cell biology



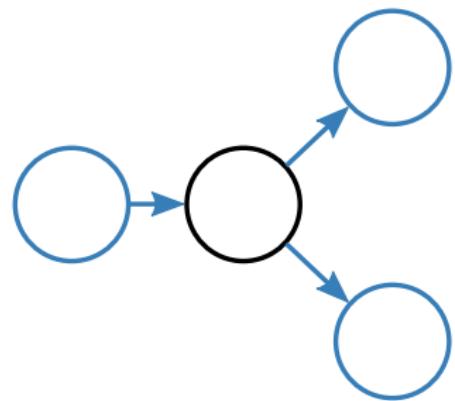
Regev *et al.* 2018

## Research questions

## Research questions

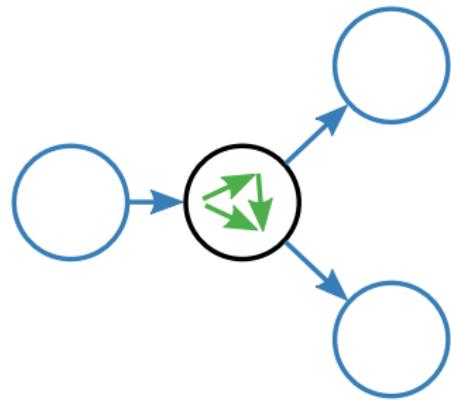


## Research questions



How do cells change over time?

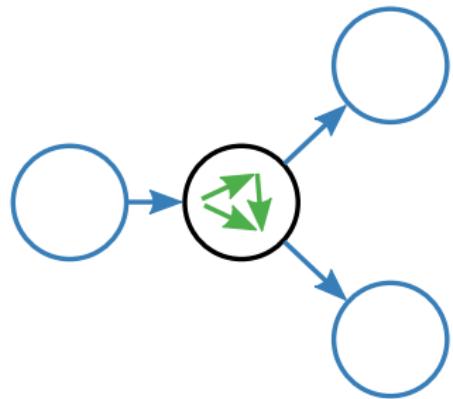
# Research questions



How do cells change over time?

What drives cells to change?

# Research questions



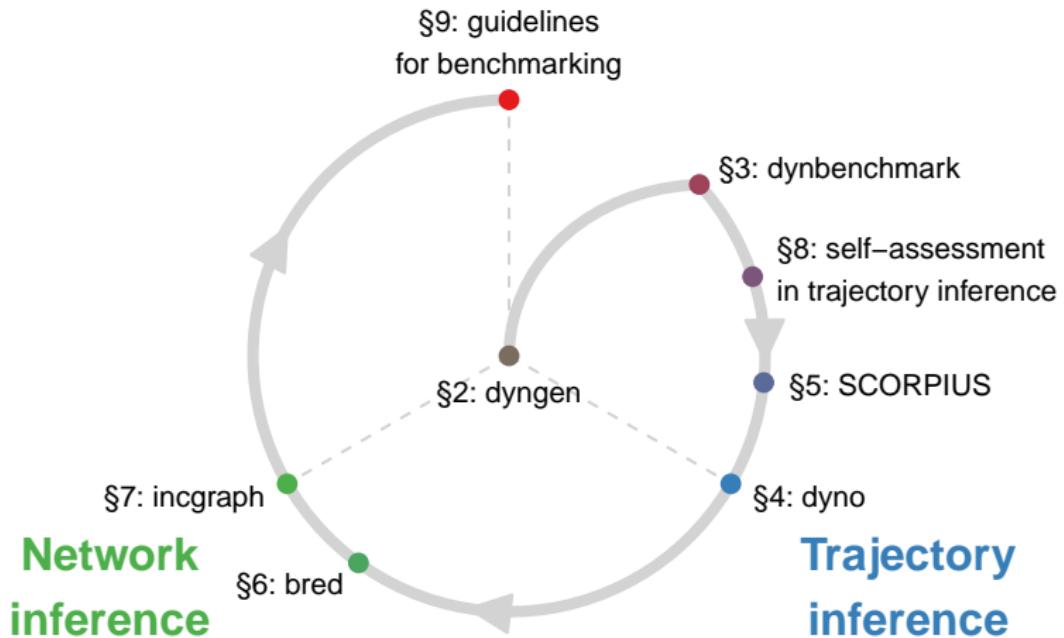
**How do cells change over time?**

→ Trajectory inference

**What drives cells to change?**

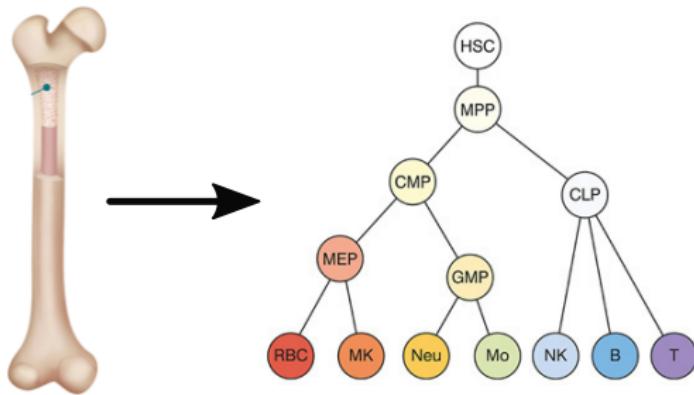
→ Network inference

## Benchmarking

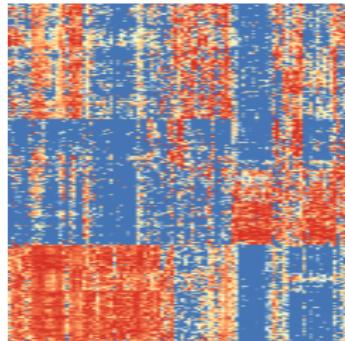


Trajectory inference

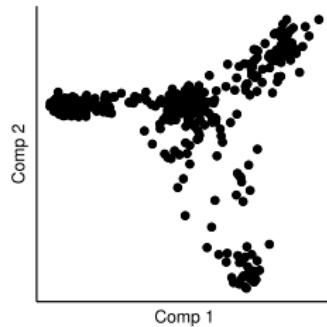
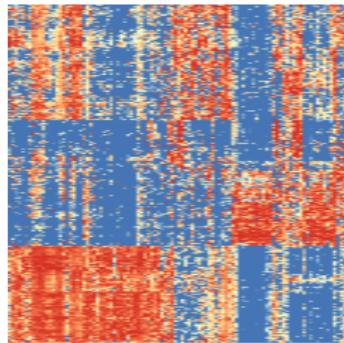
# Cells are highly dynamic entities



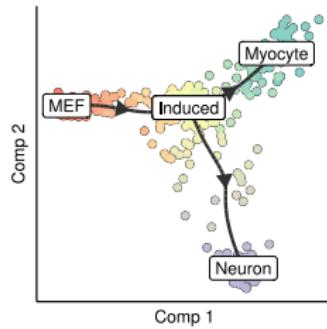
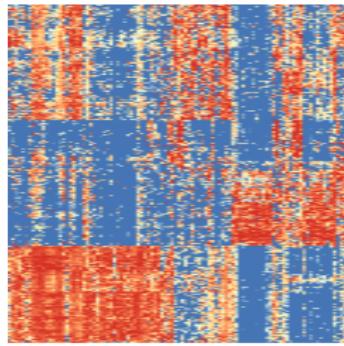
# A typical trajectory inference analysis



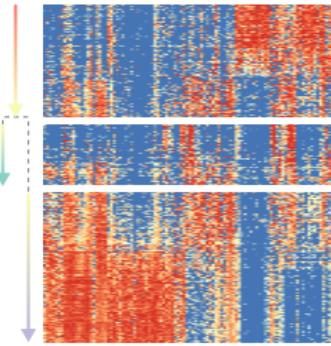
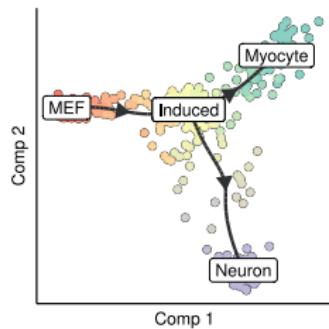
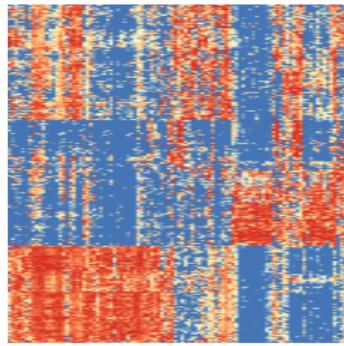
# A typical trajectory inference analysis



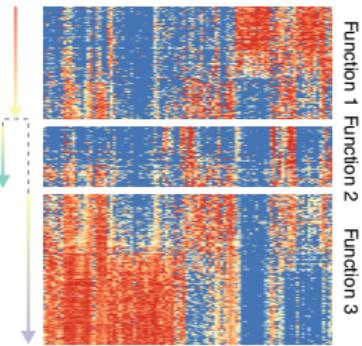
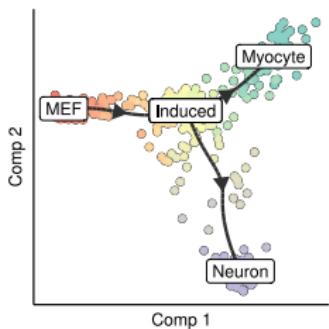
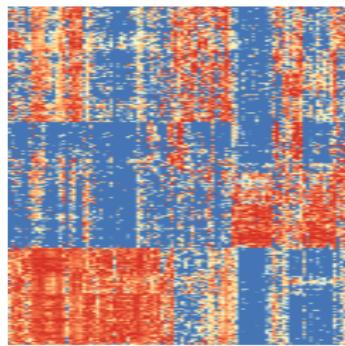
# A typical trajectory inference analysis



# A typical trajectory inference analysis

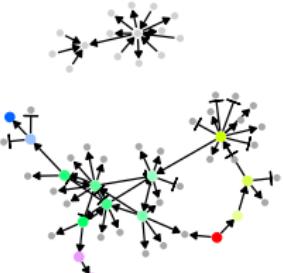


# A typical trajectory inference analysis

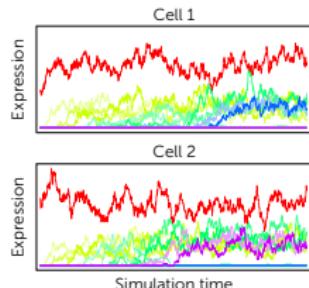


# dyngen: Benchmarking with *in silico* cells

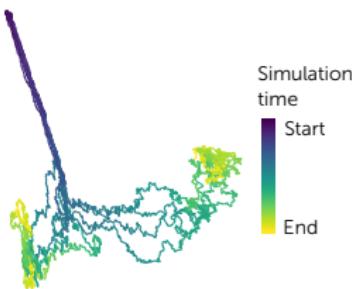
Gene regulatory network



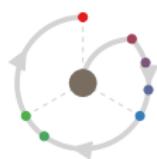
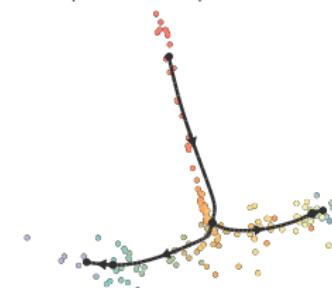
Simulate expression over time



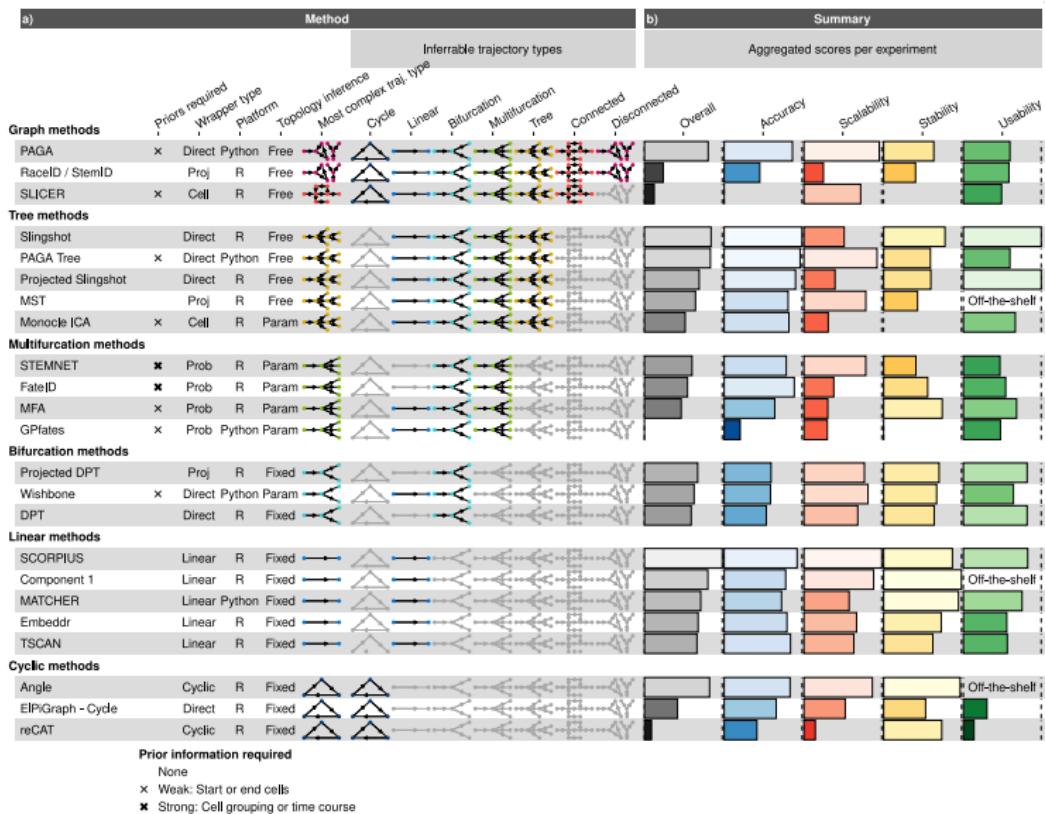
Combine simulations



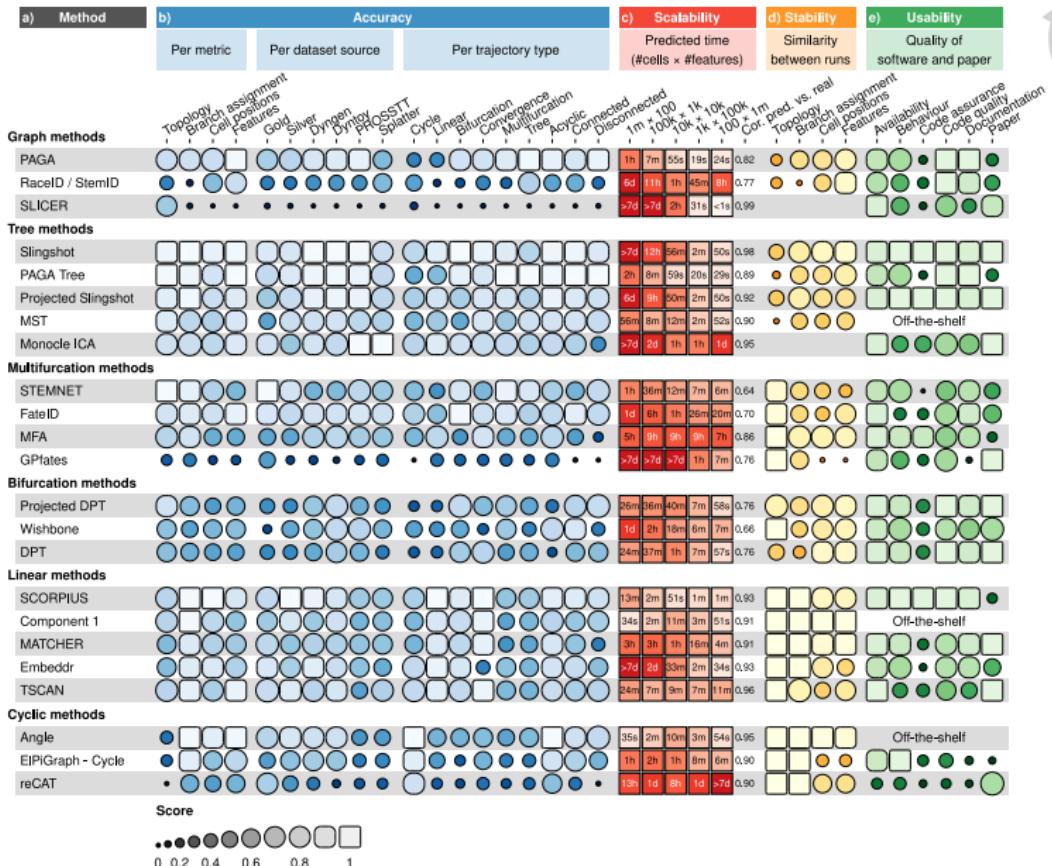
Sample cells & map to backbone



# Benchmark of 45 trajectory inference methods



## Benchmark of 45 trajectory inference methods



# Interactive guidelines



dynguidelines

Tutorial Citation

Topology  
Do you expect multiple disconnected trajectories in the data?

Scalability COMPUTED

Number of cells

Number of features (genes)

Time limit  
10s

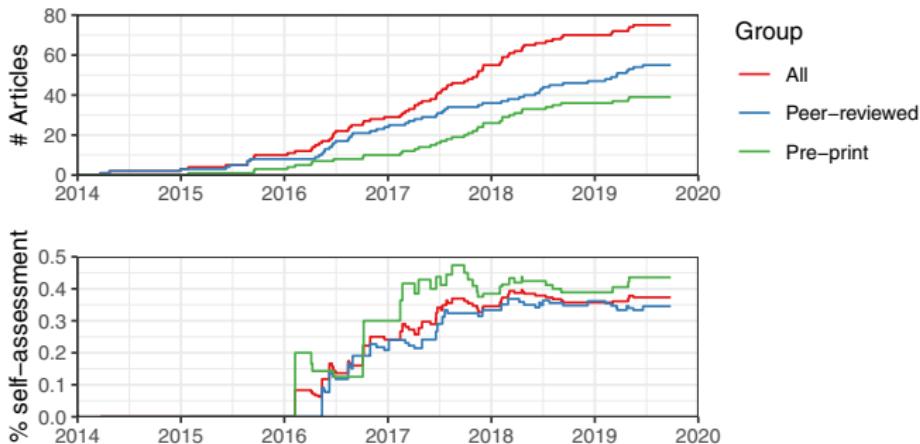
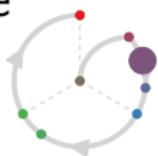
Memory limit  
10MB

Show code Show/hide columns Options

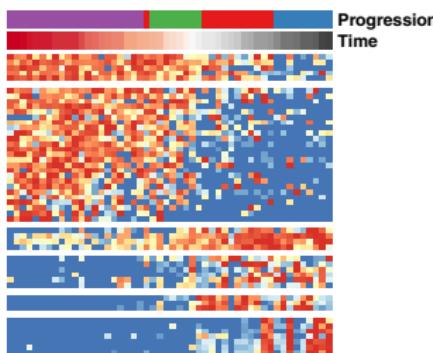
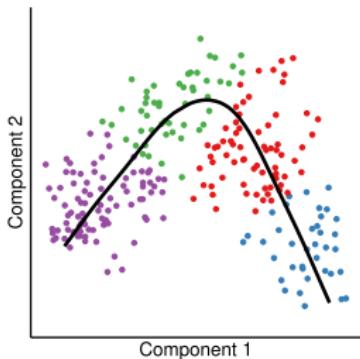
Benchmark study Evaluating methods with dyngbenchmark Part of

Lenses	Default	Summary (Fig. 2)	Method	Scalability	Stability	Usability	Accuracy	Overall	Everything
<b>Method</b>									
							<b>Accuracy</b>	<b>Scalability</b>	<b>Stability</b>
			Name	Priors			Errors	Overall #	O.T
			Singshot				100	8s	942MB
			SCORPIUS				96	3s	507MB
			Angle				92	1s	308MB
			PAGA				59	15s	559MB
			Embeddr				89	5s	581MB
			MST				69	4s	572MB
			Waterfall				69	5s	369MB
			TSCAN				68	5s	476MB
			Component 1				67	1s	516MB
			SLICE				63	16s	713MB
			Monocle DDRTree				62	41s	647MB
			EPIGraph Linear				61	1m	573MB

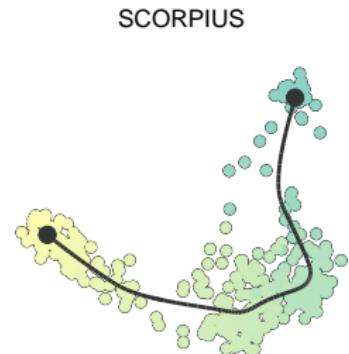
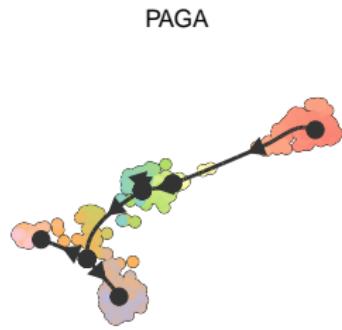
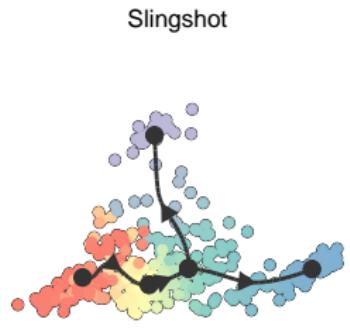
# Perspective: self-assessment in trajectory inference



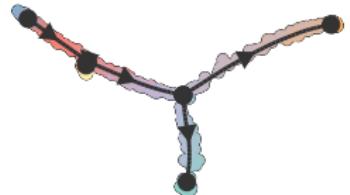
# SCORPIUS: Linear trajectory inference



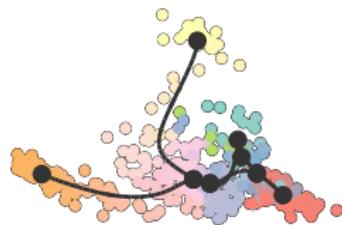
# dyno: A toolkit for inferring and interpreting trajectories



Monocle DDRTree



pCreode



# Network inference

Cellular dynamics is (i.a.) driven by gene regulation



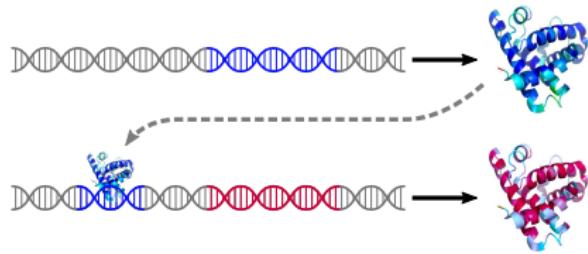
Cellular dynamics is (i.a.) driven by gene regulation



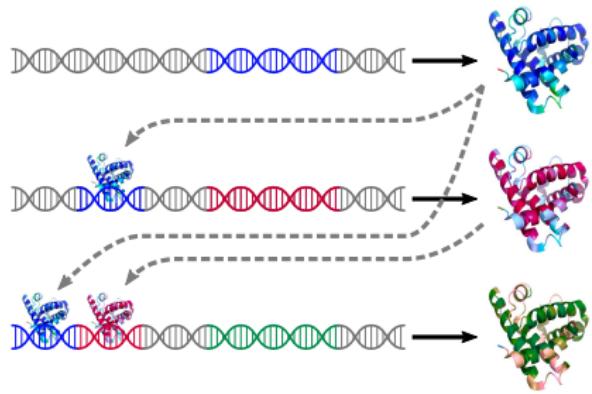
Cellular dynamics is (i.a.) driven by gene regulation



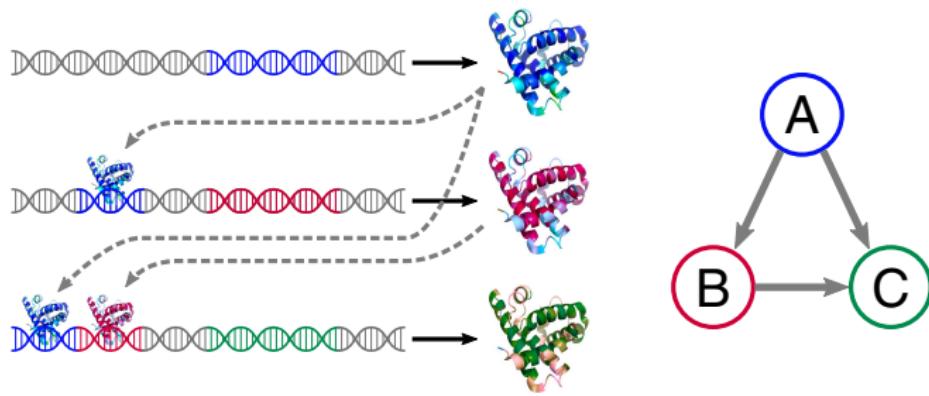
Cellular dynamics is (i.a.) driven by gene regulation



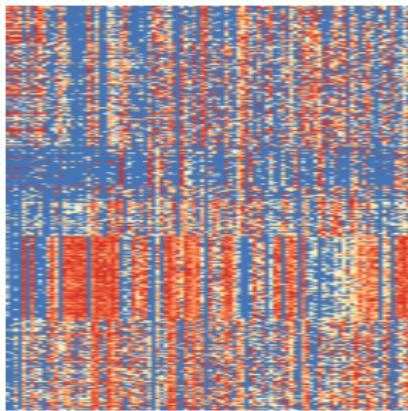
# Cellular dynamics is (i.a.) driven by gene regulation



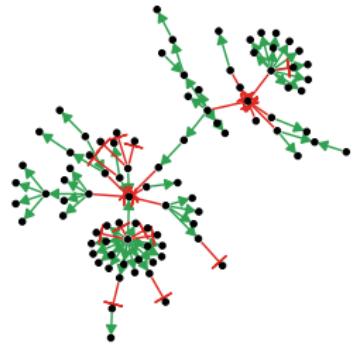
# Cellular dynamics is (i.a.) driven by gene regulation



# Classical network inference

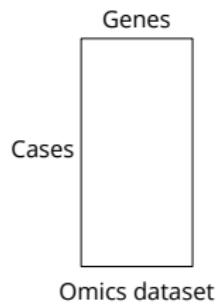


Network  
inference

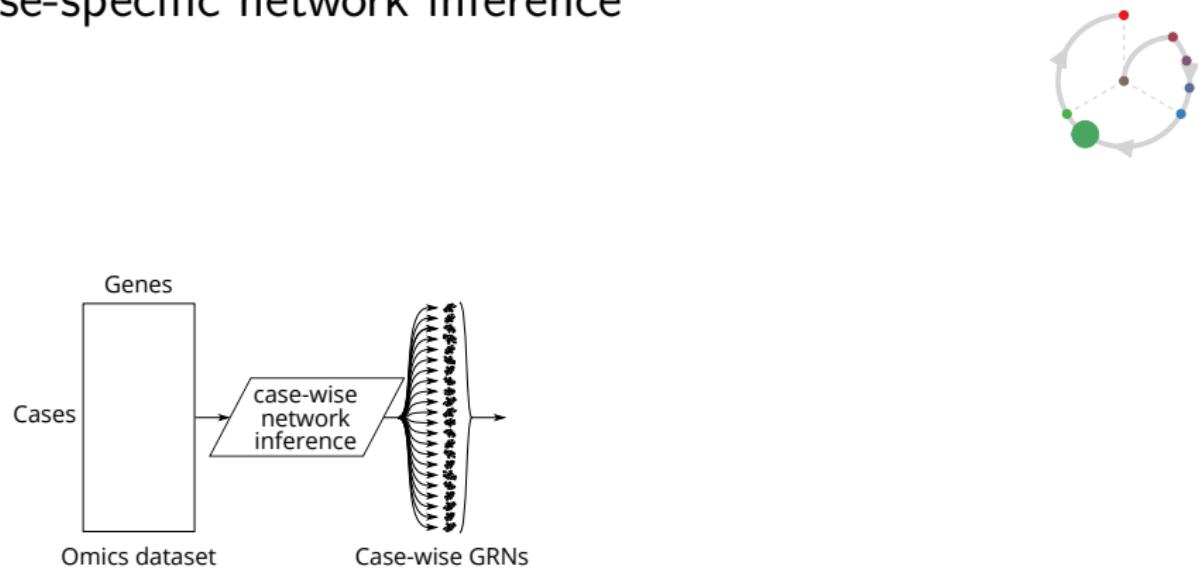


Interaction type    → Promotes    → Represses

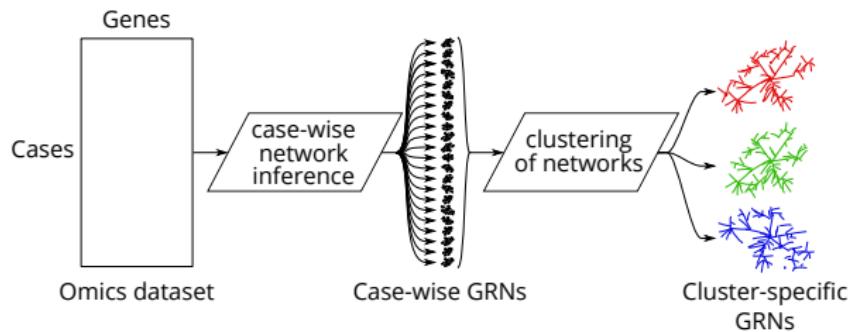
# Case-specific network inference



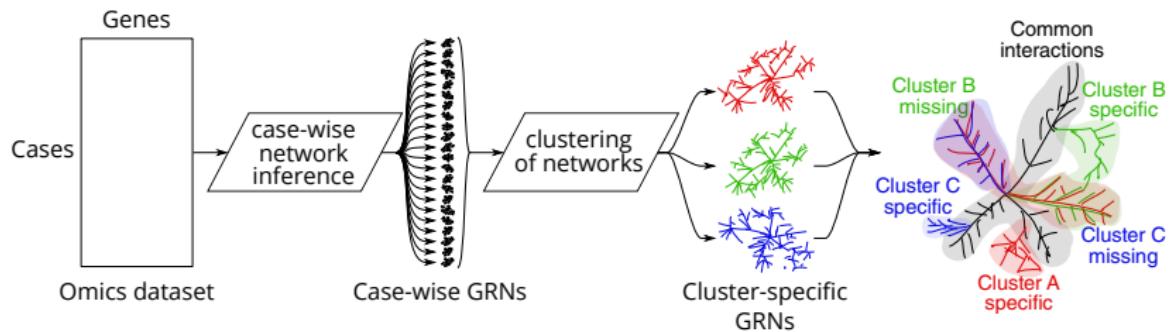
# Case-specific network inference



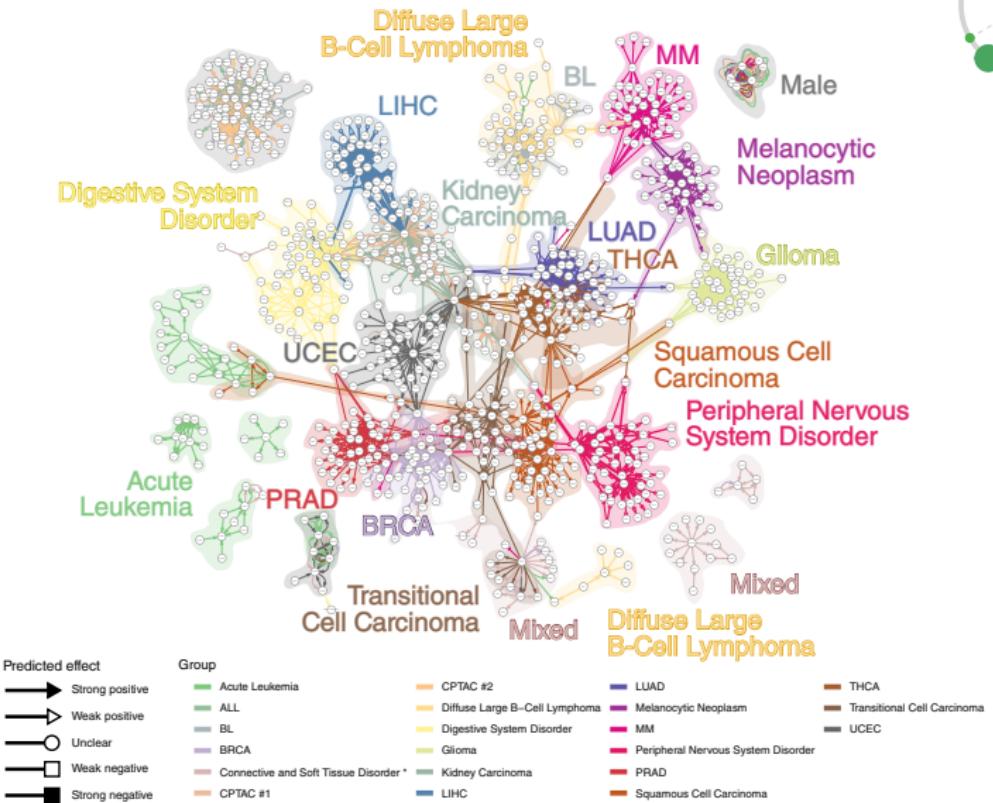
# Case-specific network inference



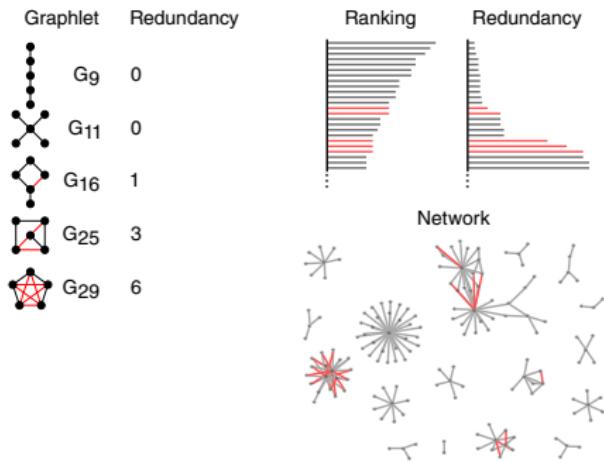
# Case-specific network inference



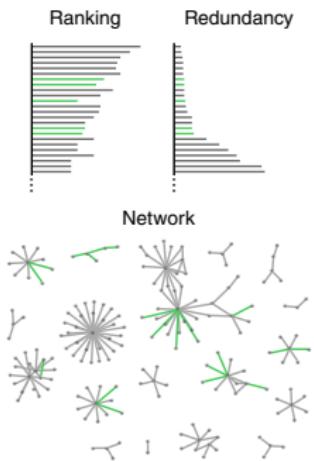
# bred applied to TCGA



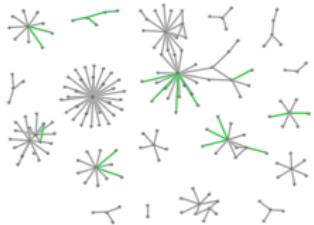
# incgraph: Optimising regulatory networks



Rerank to lower redundancy

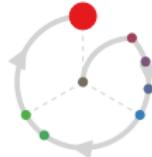


Network



# Benchmarking

# Essential guidelines for benchmarking computational tools



1. Define the purpose and scope of the benchmark.
2. Include all relevant methods.
3. Select (or design) representative dataset.
4. Choose appropriate parameter values and software versions.
5. Evaluate methods according to key quantitative performance metrics.
6. Evaluate secondary measures including computational requirements, user-friendliness, installation procedures, and documentation quality.
7. Interpret results and provide recommendations from both user and method developer perspectives.
8. Publish results in an accessible format
9. Design the benchmark to enable future extensions.
10. Follow reproducible research best practices, by making code and data publicly available.

## Practical implications



## Practical implications

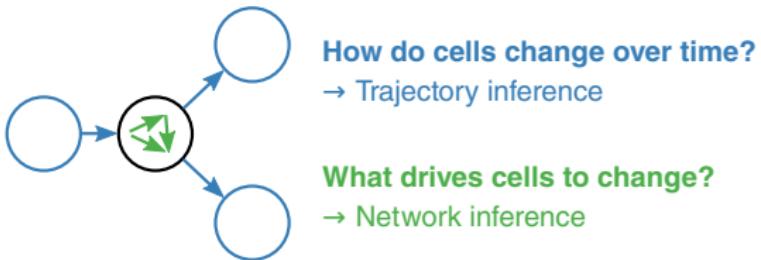


- Fundamental research in single-cell analyses

# Practical implications



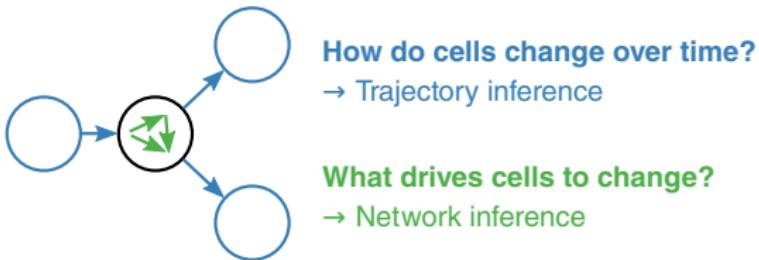
- Fundamental research in single-cell analyses



# Practical implications



- Fundamental research in single-cell analyses

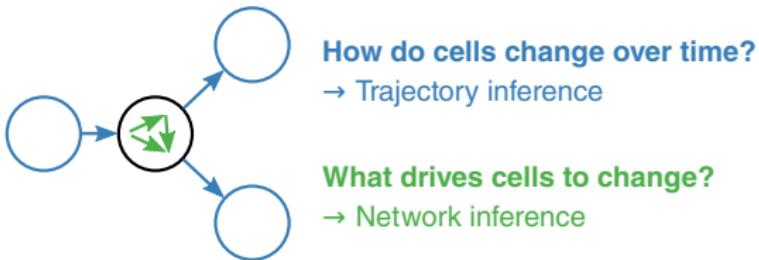


- Benchmarking

# Practical implications



- Fundamental research in single-cell analyses



- Benchmarking
- R & tidyverse