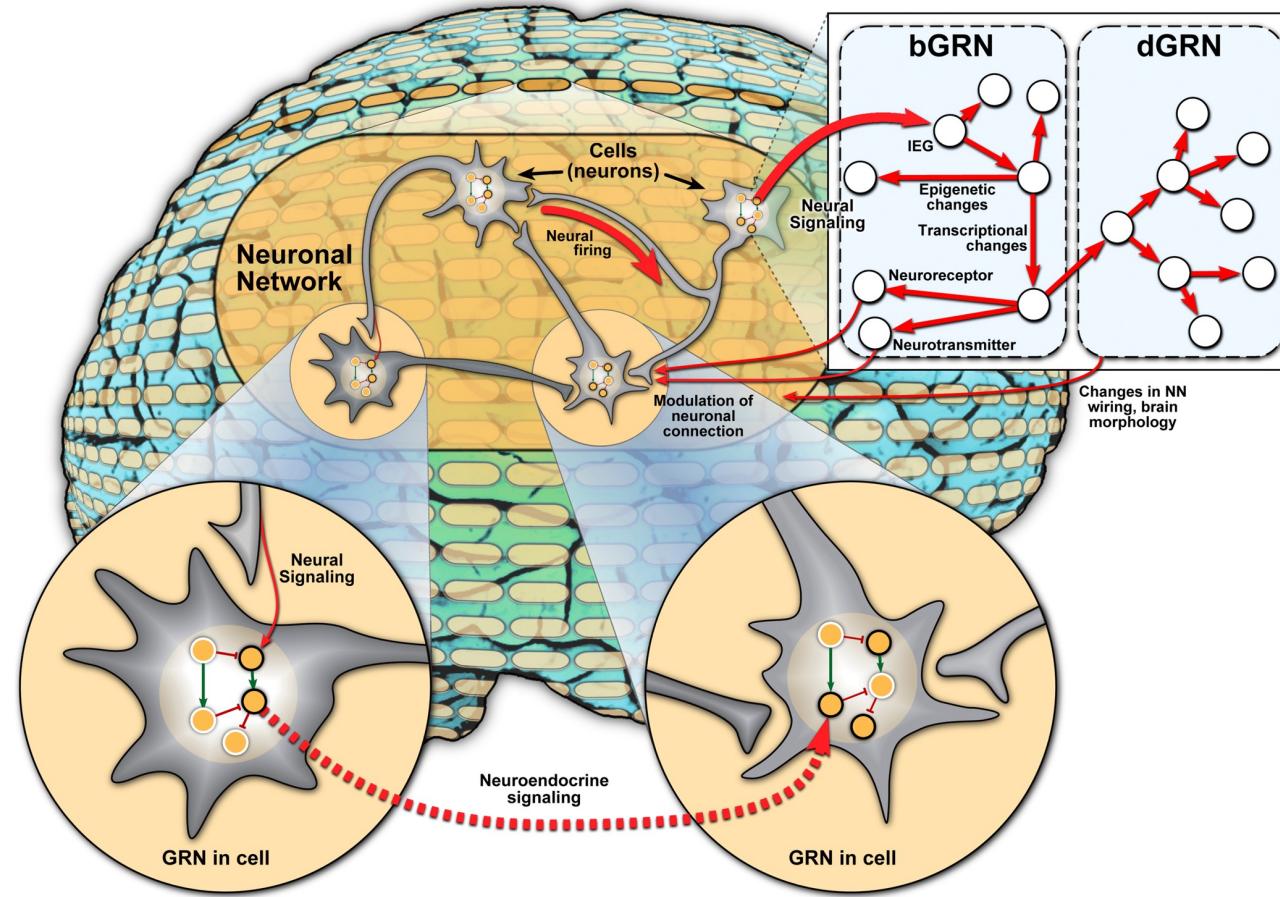
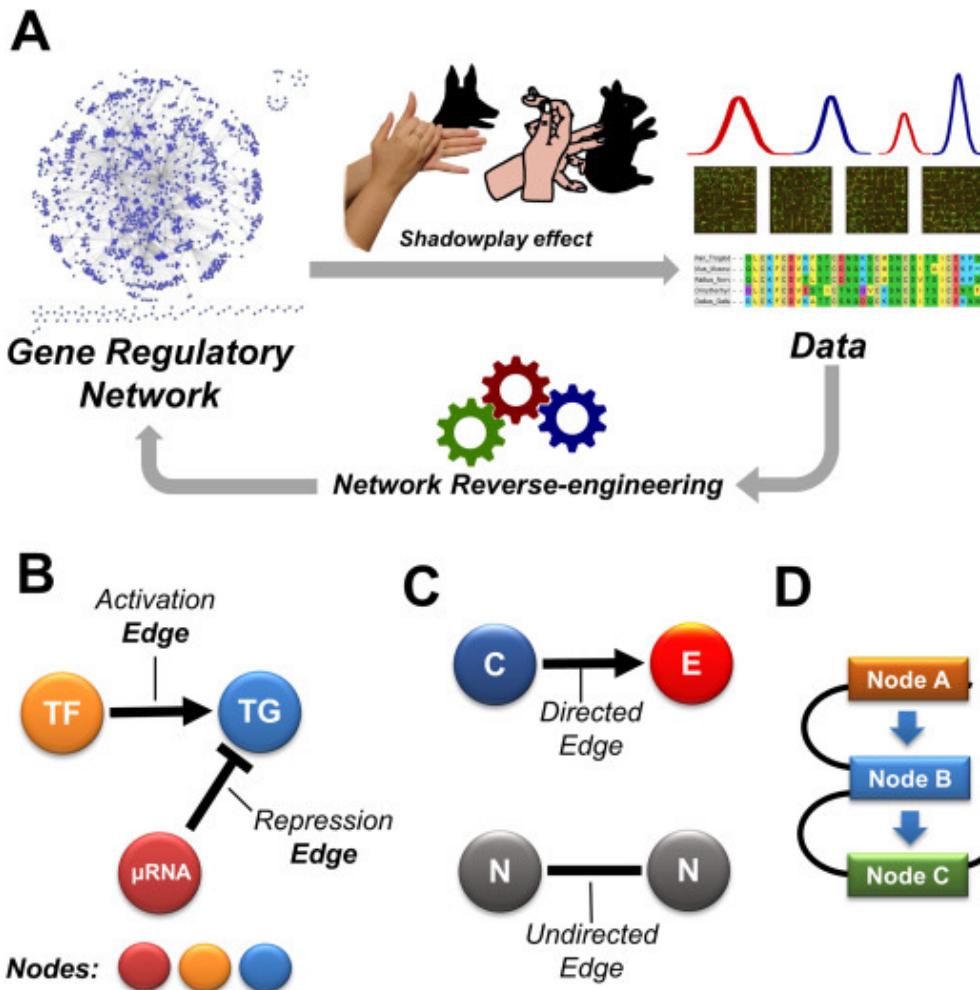


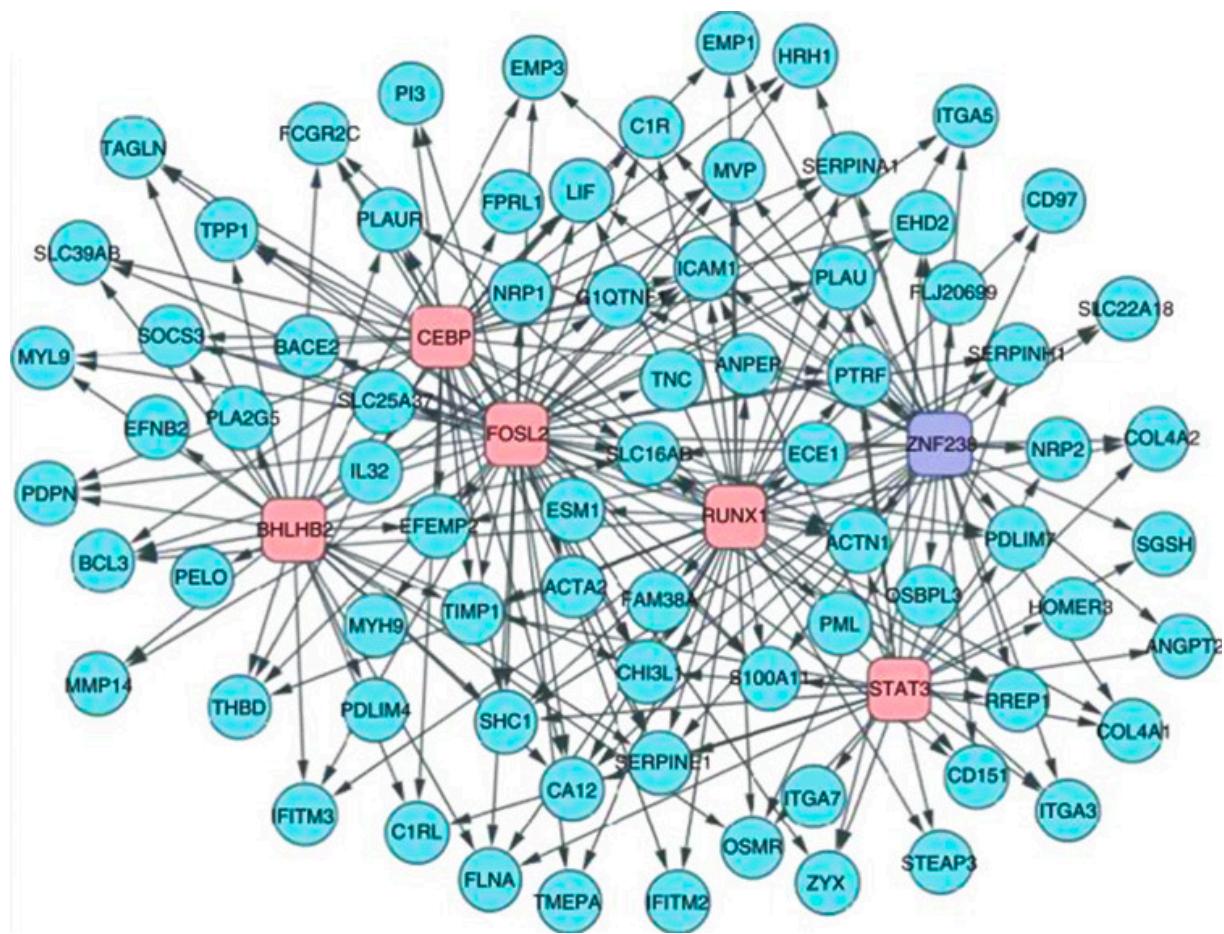
# Gene Regulatory Network Inference in Fetal Brain



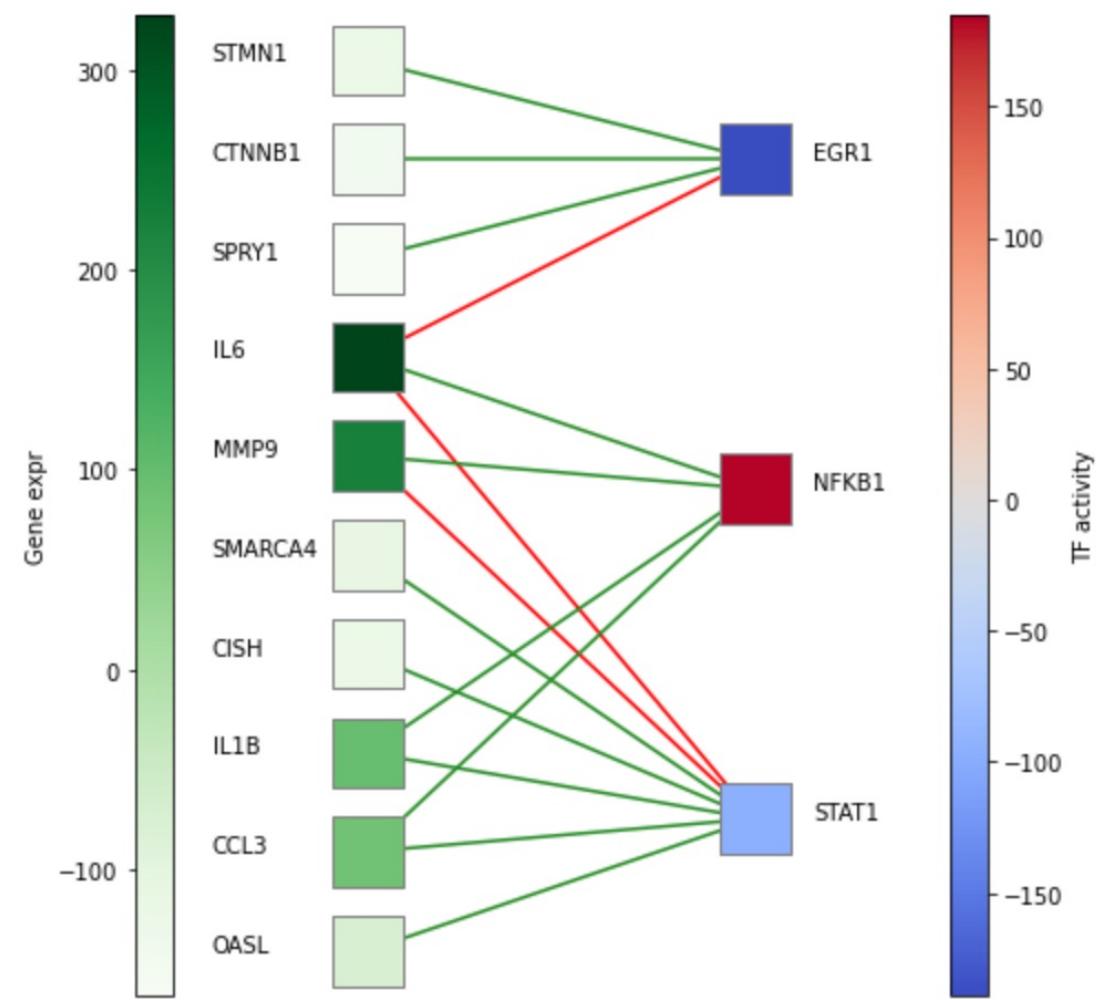
# Gene regulatory network Inference



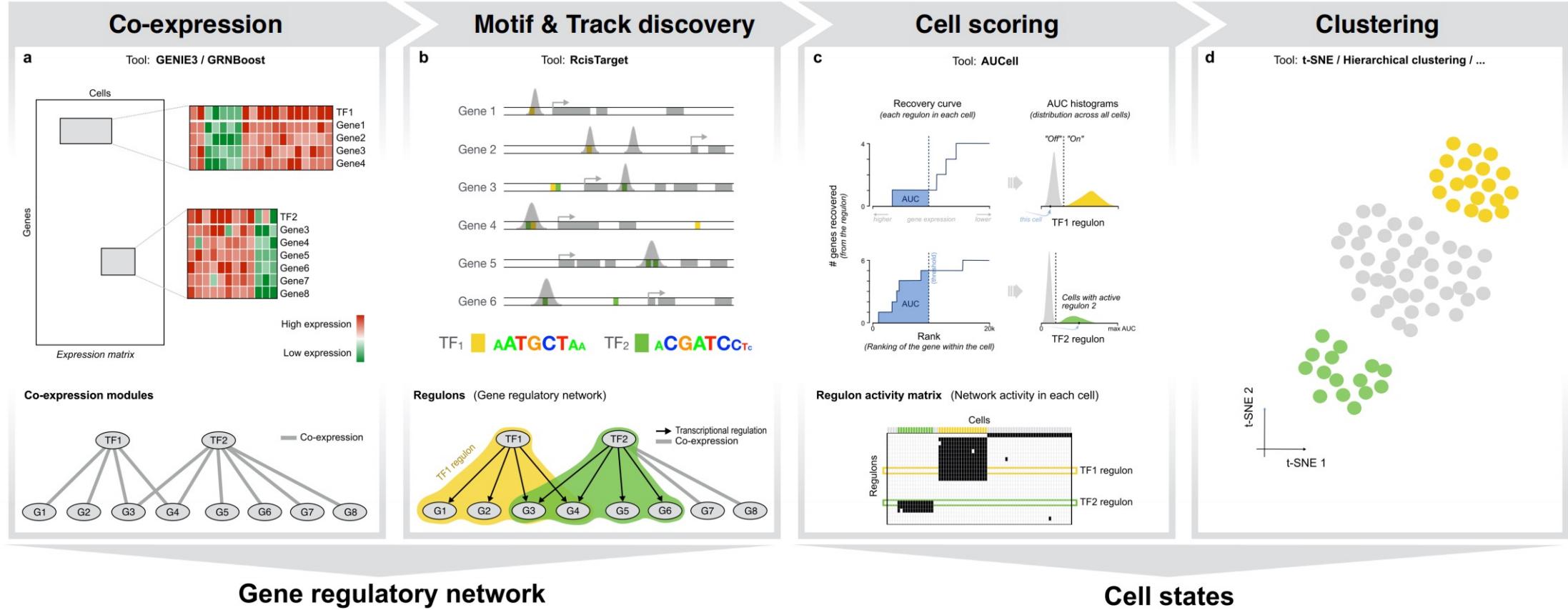
# TF Activity: a primer



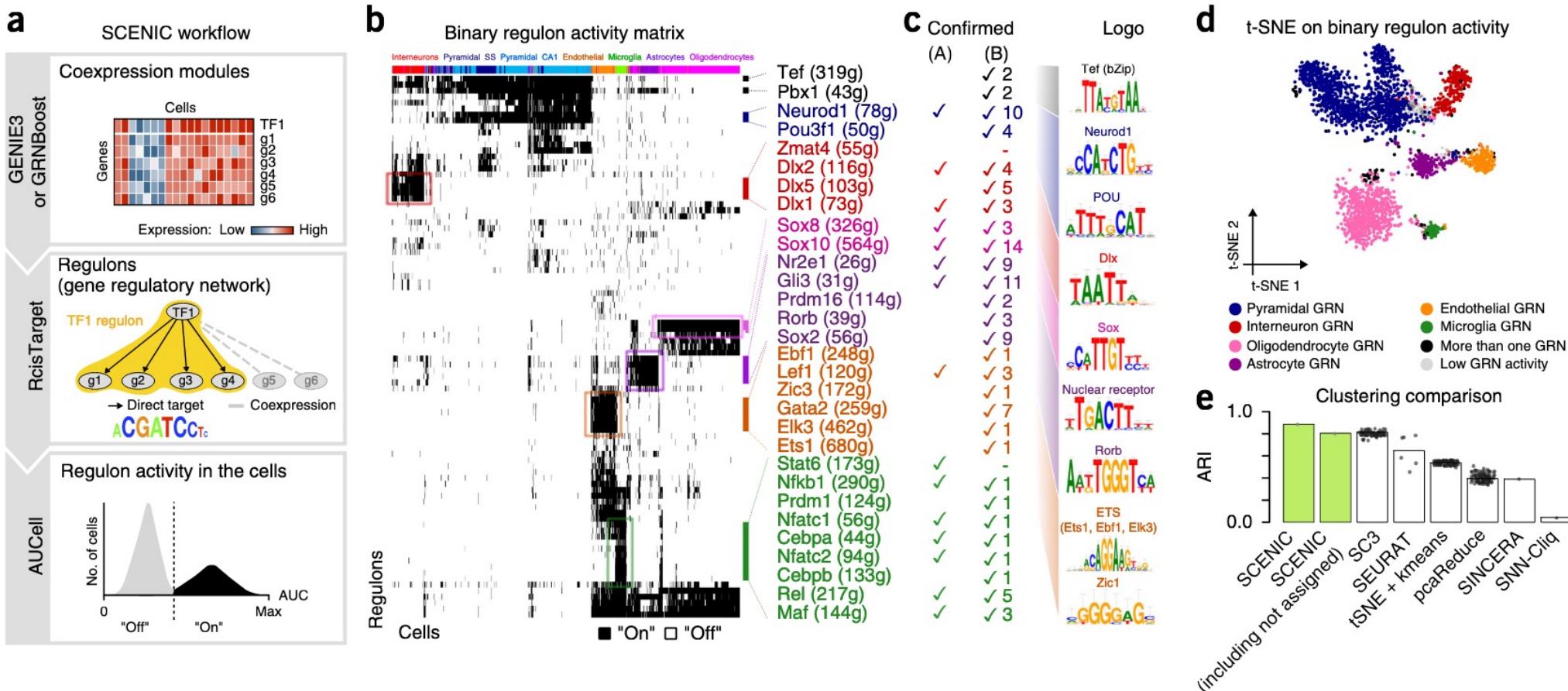
Footprint-based enrichment analysis



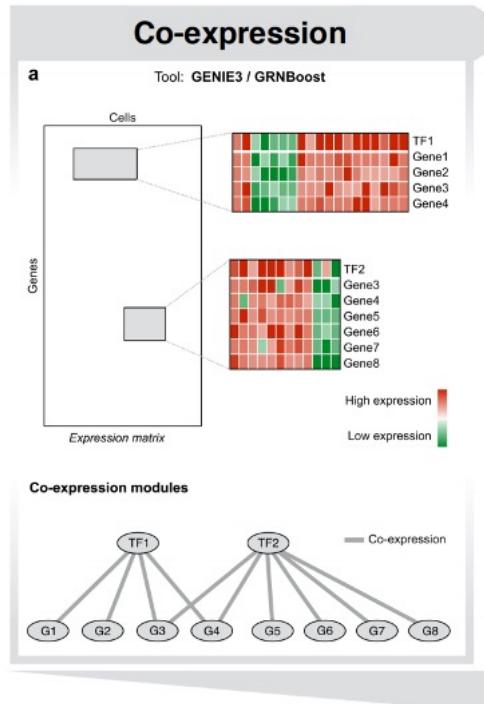
# SCENIC PROTOCOL



# SCENIC to cluster cells based on TF Activity



# SCENIC –STEP1 – Inferring the GRN



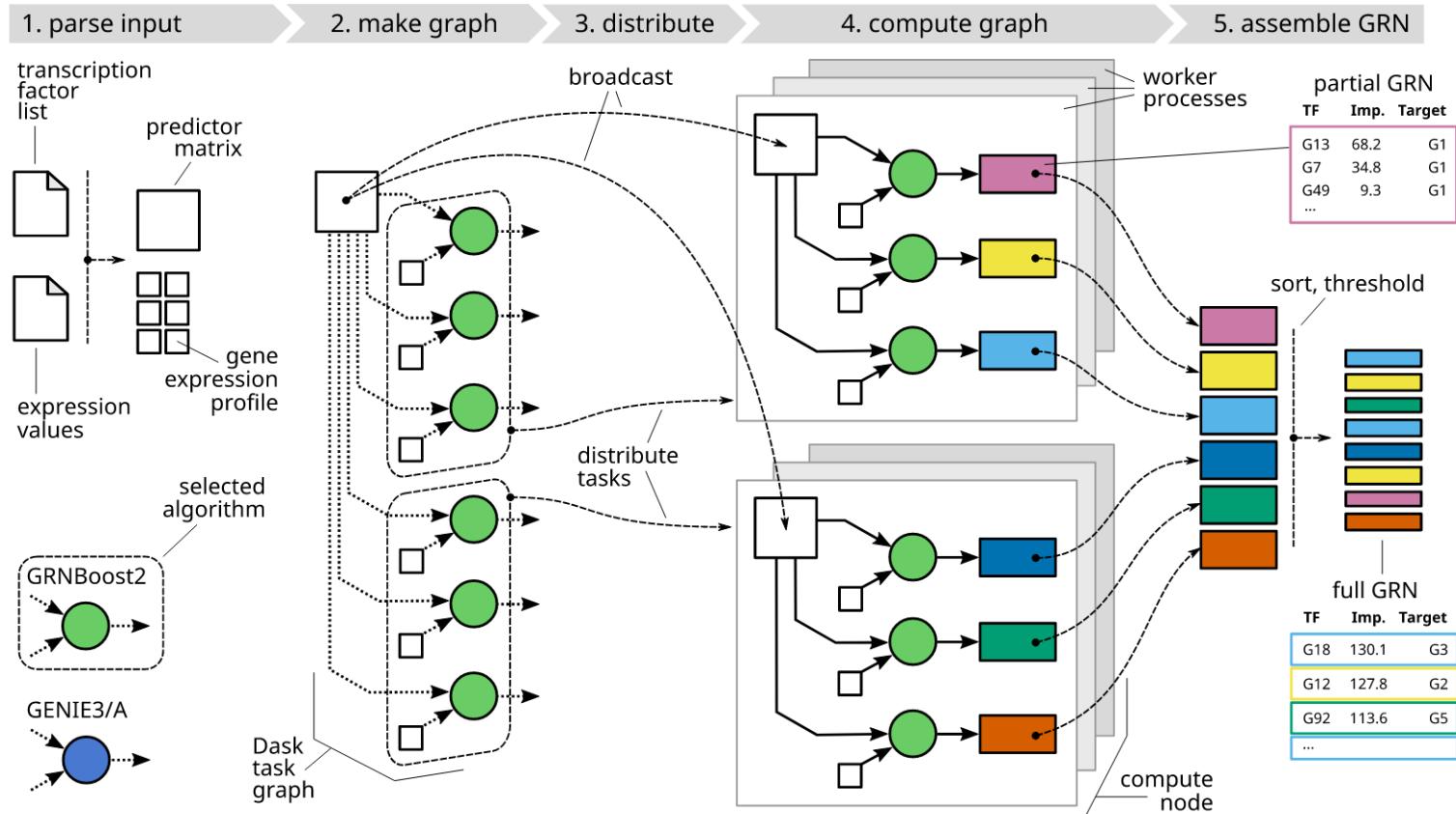
## Example command with GRNBoost

```
pyscenic grn \
--num_workers 20 \
--output adj.tsv \
--method grnboost2 \
PBMC10k_filtered.loom \
hs_hgnc_tfs.txt
```

## You will need:

- A loom file storing all the data (the grn inference algorithms **loves** raw counts)
- A file storing TFs, one per line
- A LOT OF MEMORY
- Some time :)

# GENIE3 and GRNBOOST with Arboreto

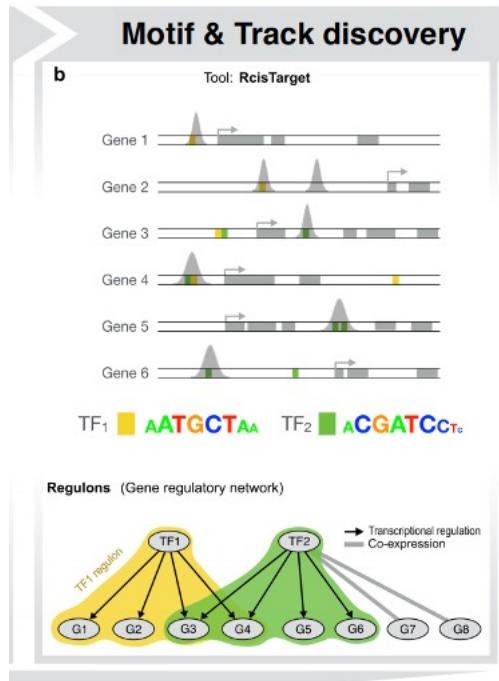


**Figure 1:** The Arboreto distributed GRN inference architecture.

<https://academic.oup.com/bioinformatics/article/35/12/2159/5184284?login=true>

# STEP2 – Prune GRN using RCisTarget

## Example Command



```
pyscenic ctx \
adj.tsv \
hg38_refseq-r80_10kb_up_and_down_tss.mc9nr.feather \
--annotations_fname motifs-v9-nr.hgnc-m0.001-o0.0.tbl \
--expression_mtx_fname PBMC10k_filtered.loom \
--mode "dask_multiprocessing" \
--output reg.csv \
--num_workers 20 \
--mask_dropouts
```

### You will need:

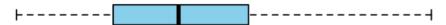
- the loom file used before
- A .tbl file storing motif -> TF association
- A feather file storing all the motif
- The grn inferred with pyscenic ctx

Create RcisTarget Database

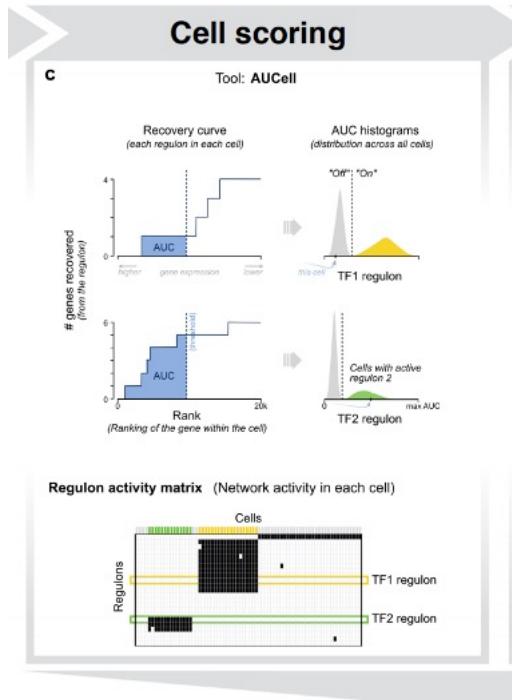
[https://github.com/aertslab/create\\_cisTarget\\_databases](https://github.com/aertslab/create_cisTarget_databases)

Everything you need to run it out of the box is here:

<https://resources.aertslab.org/cistarget/>



# STEP 3 - AUCELL

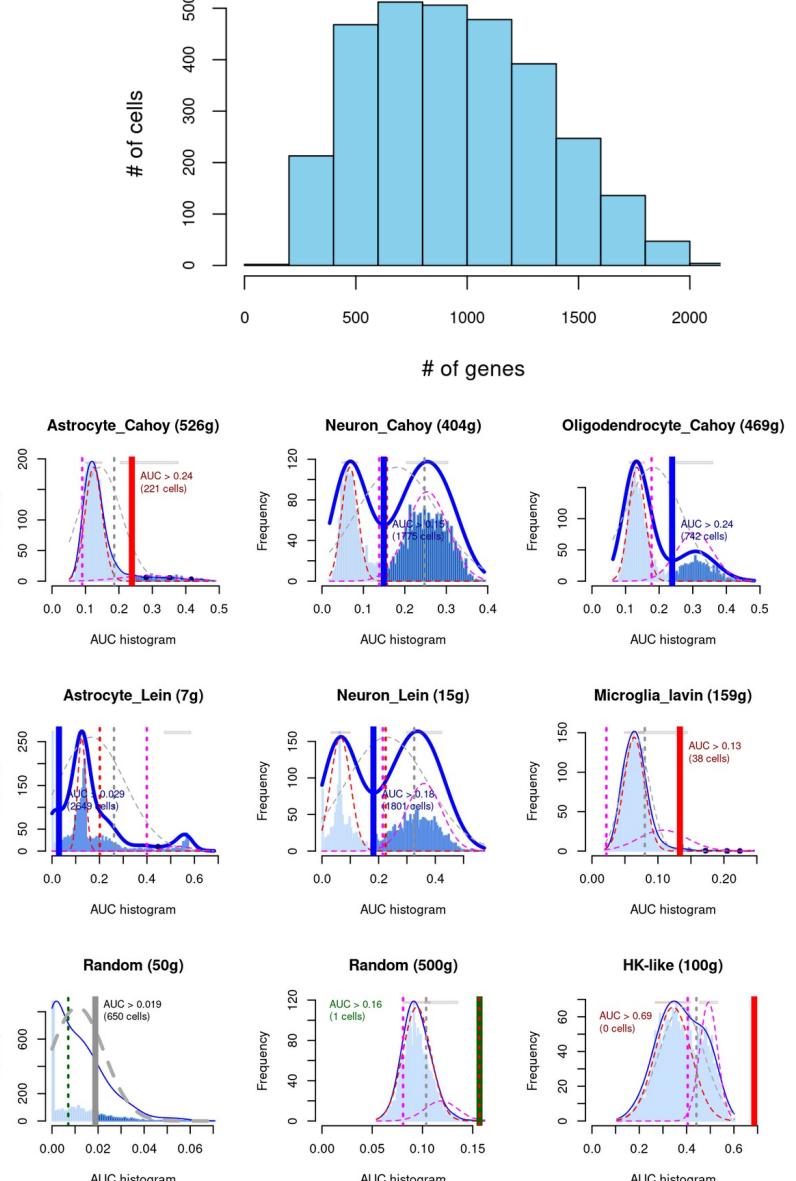


## Example Command

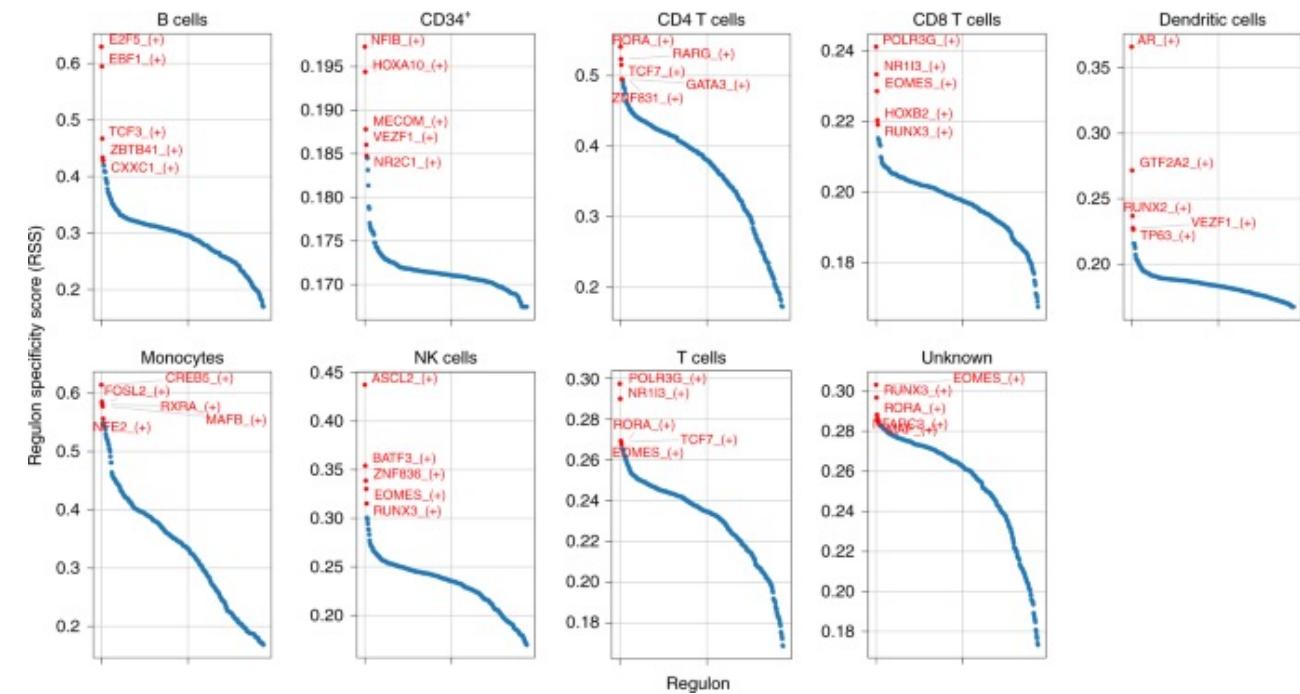
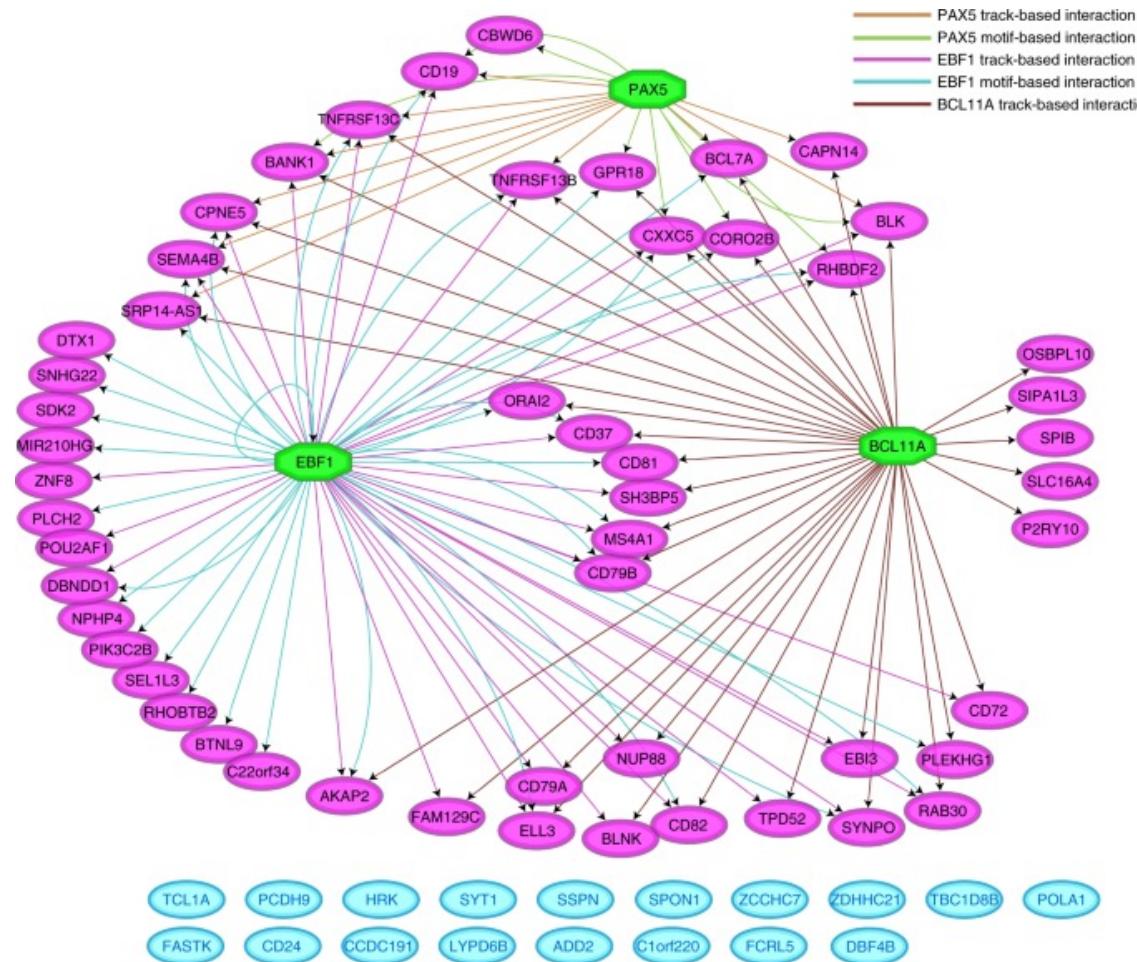
```
pyscenic aucell \
PBMC10k_filtered.loom \
reg.csv \
--output PBMC10k_SCENIC.loom \
--num_workers 20
```

## You will need:

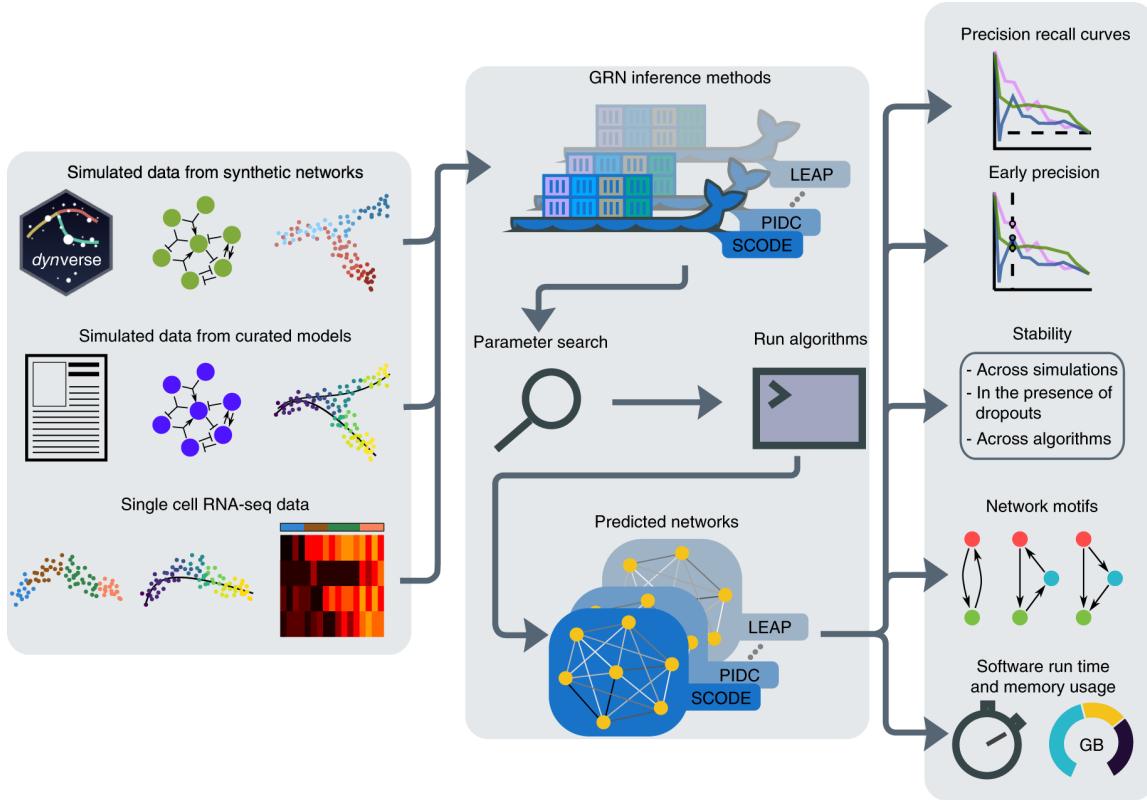
- The initial loom file
- The regulons you found in the previous step
- A FILTERING THRESHOLD (missing in the command)



# Post Processing



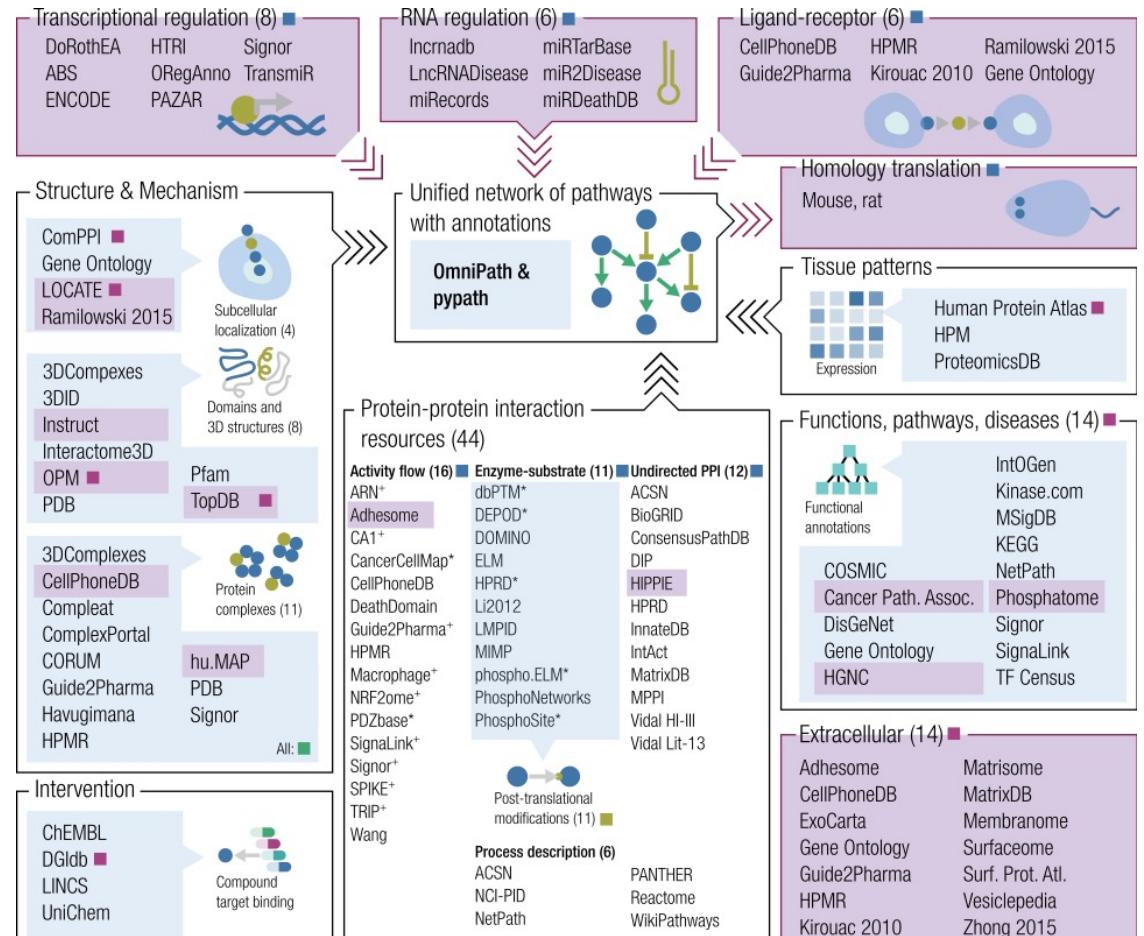
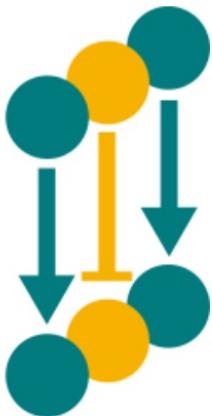
# Benchmark of GRN Inference Methods



# Caveats in GRN Inference

- The higher the quality of cells, the better.
- If you don't have a ground truth, it's difficult to evaluate if the GRN is working
- If you have a developmental process and many heterogeneous cells, it's gonna be tough
- If you have your own motif database for your condition, filtering false
- It is computationally intensive
- GRNs from single cell are still very far from reality and close to random, so consider making meta-cells and use them as 'pseudo-bulk' replicates
- You will need to run the algorithms several times and grab a consensus, is a stochastic process
- GRN can rarely infer inhibition, activation are easier to find
- Self regulatory loops and indirect regulations are still very far to be solved

# How to Retrieve a Ground Truth?



## Availability:

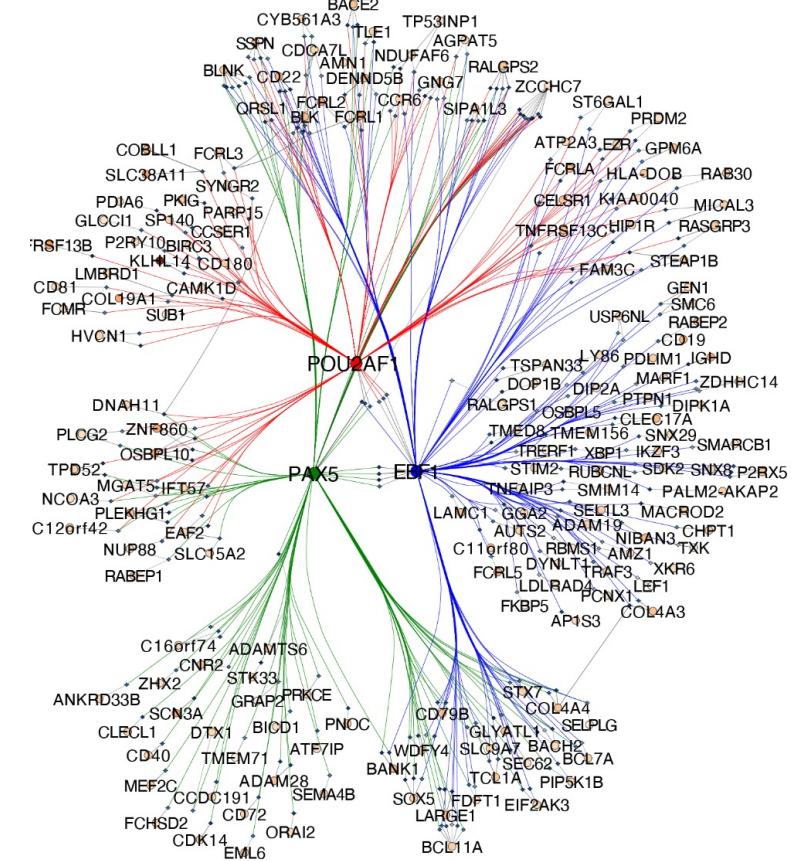
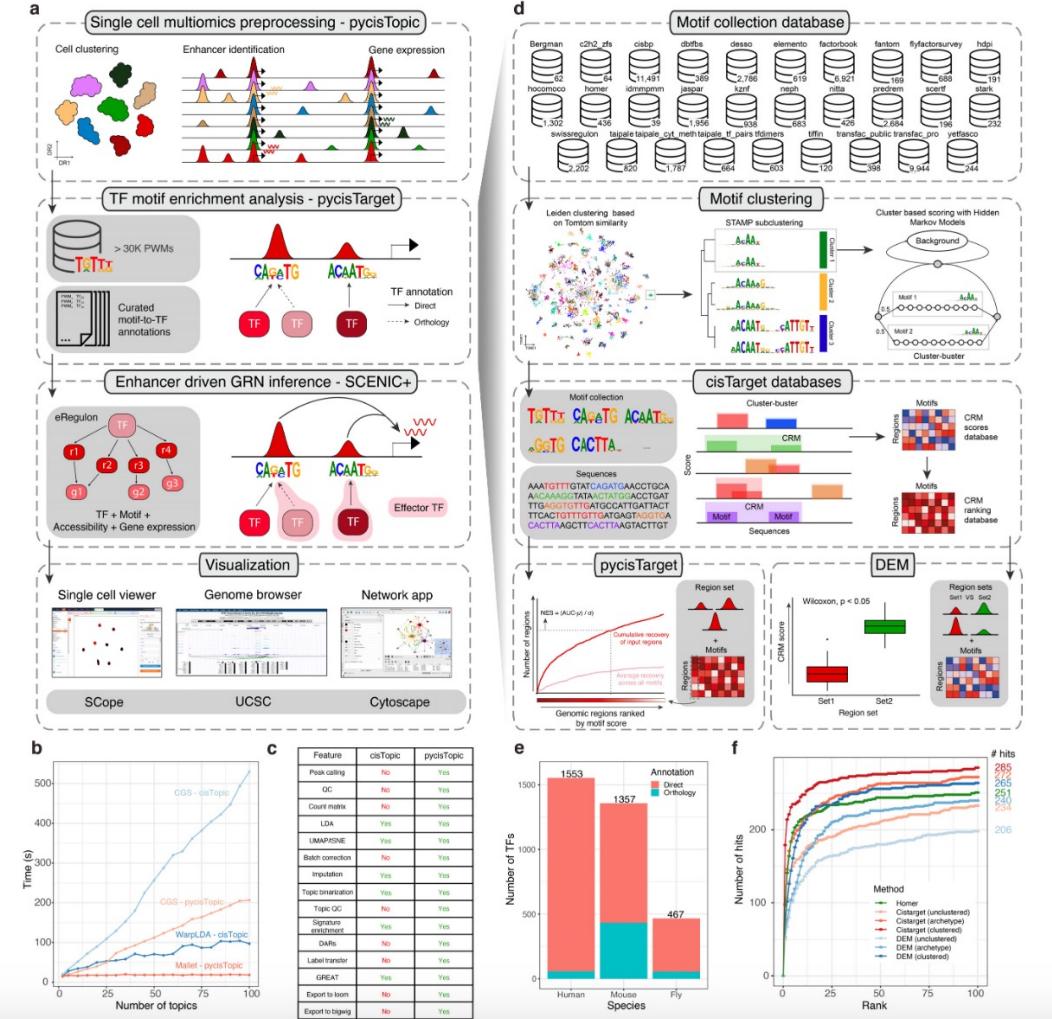
pypath.main, omnipathdb.org/interactions

pypath.ptm, omnipathdb.org/ptms

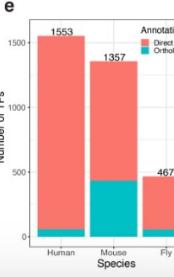
pypath.annot, pypath.intercell, omnipathdb.org/annotations, omnipathdb.intercell

pypath.complex, omnipathdb.org/complexes

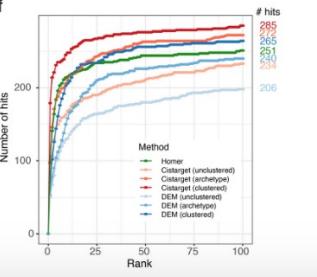
# Moving to Multimodal Data: ScenicPlus



c

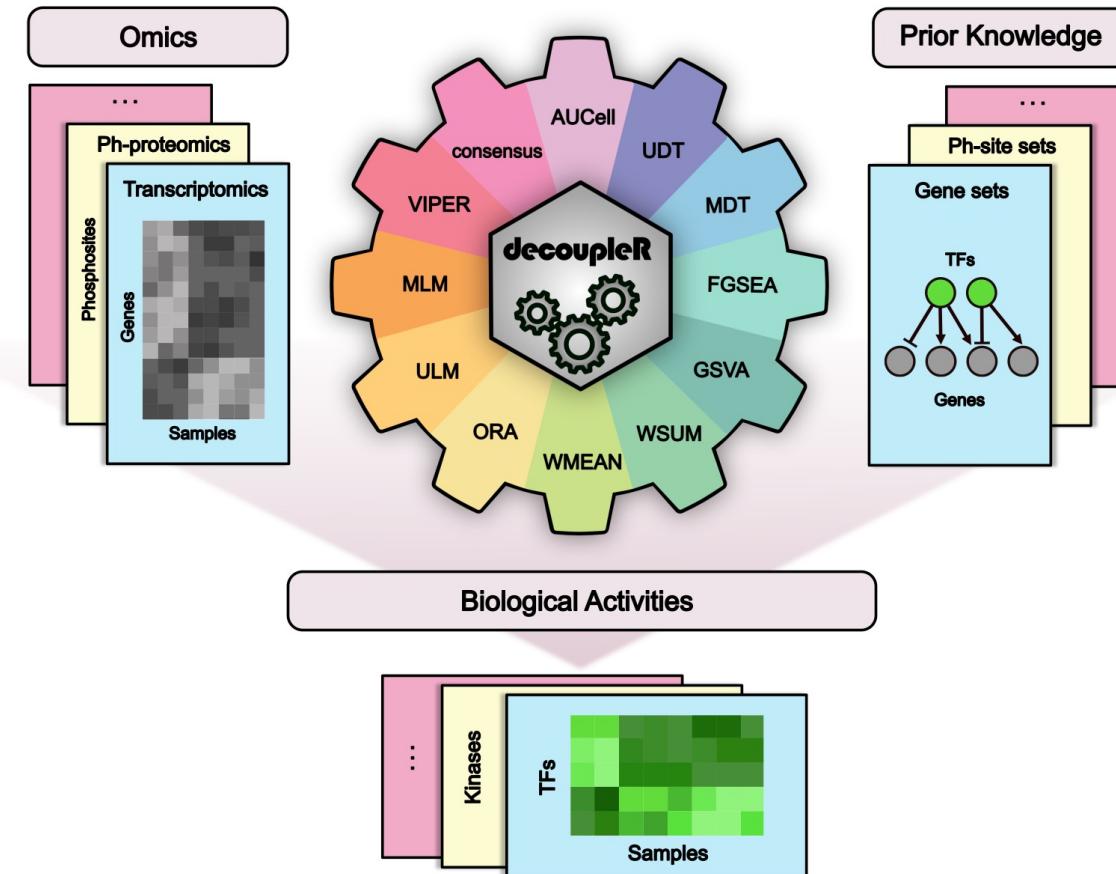


e



f

# Tf Activity Methods –all-in-one: DecoupleR



# What we will do

- Rebuild a ground truth Gene Regulatory network using OmniPath
- Test the activity of known regulons in a scRNA-Seq cortical brain development dataset using 3 different methods
- Compare it with a GRN inferred from the same data
- Check whether we can compute a umap based on the regulon activity and find some useful insights (spoiler: no)