ISMB/ECCB 2025 Tutorial VT-8
Generative AI for Single-Cell Perturbation Modeling:
Theoretical and practical considerations

Introduction to perturbation modelling for single-cell technologies

George Gavriilidis
ggeorav@certh.gr

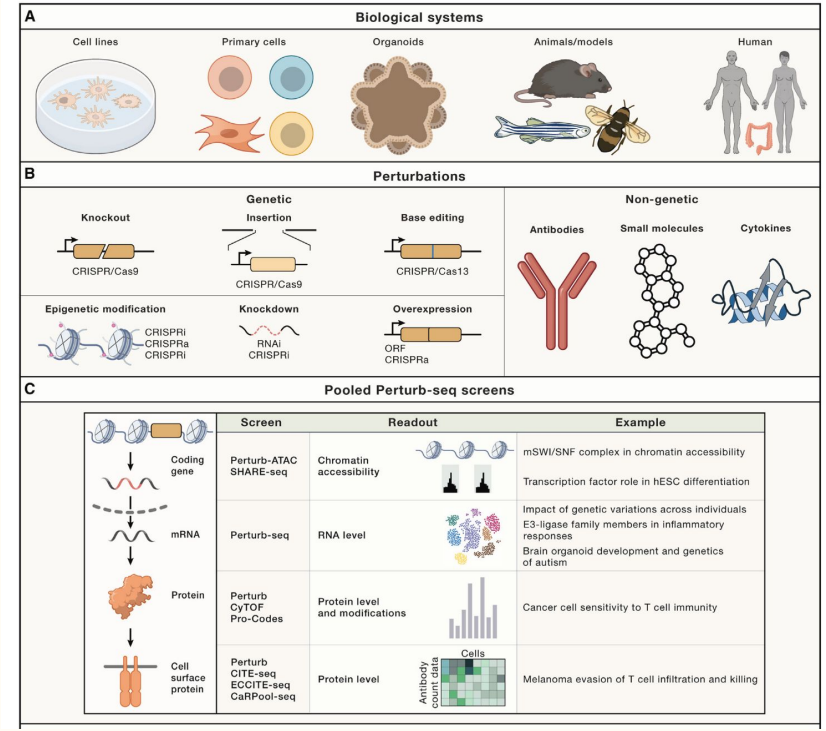
CERTH
CENTRE FOR
RESEARCH & TECHNOLOGY
HELLAS

INAB
INSTITUTE OF APPLIED BIOSCIENCES

Perturbation modelling in single-cell biology

- Single-cell technologies = unprecedented resolution into cell physiology / cell-cell communication / gene regulation

- **Perturbations:** *Growth factors/drugs/CRISPR Knock-outs* in single-cell experiments
  - Changing pathways/transcription factors?
  - Which cells are mostly affected?
  - ***Predict perturbed cell states in other datasets?***

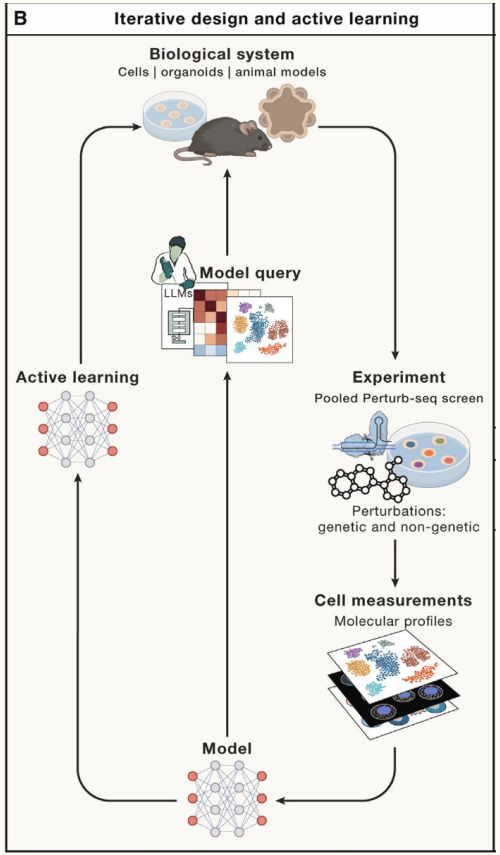


From *"chip"* to *"lab-bench"* to *"clinical bedside"*

Vision: Accelerate therapeutic discoveries from *in silico* to *in vitro* to in *vivo/clinical*

Applications: *Multiomics, Drug Repurposing/Repositioning, Drug Discovery, Biomarker Research, Immunophenotyping…*

Active Learning/Lab-on-a-loop



Perturbation modelling tools

Currently approximately 70 to 80 tools and growing..

Our recent review captures ~40 of them..

COMPUTATIONAL AND STRUCTURAL BIOTECHNOLOGY JOURNAL

OPEN ACCESS

About Sections Articles Publish Contact

Search for...

MINI-REVIEW · Volume 23, P1886-1896, December 2024 · *Open Access*

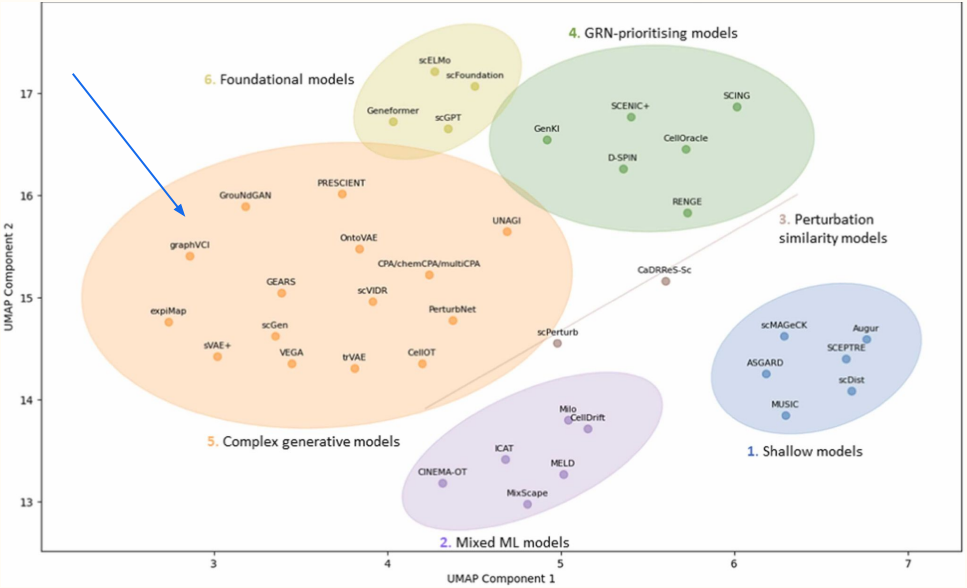
A mini-review on perturbation modelling across single-cell omic modalities

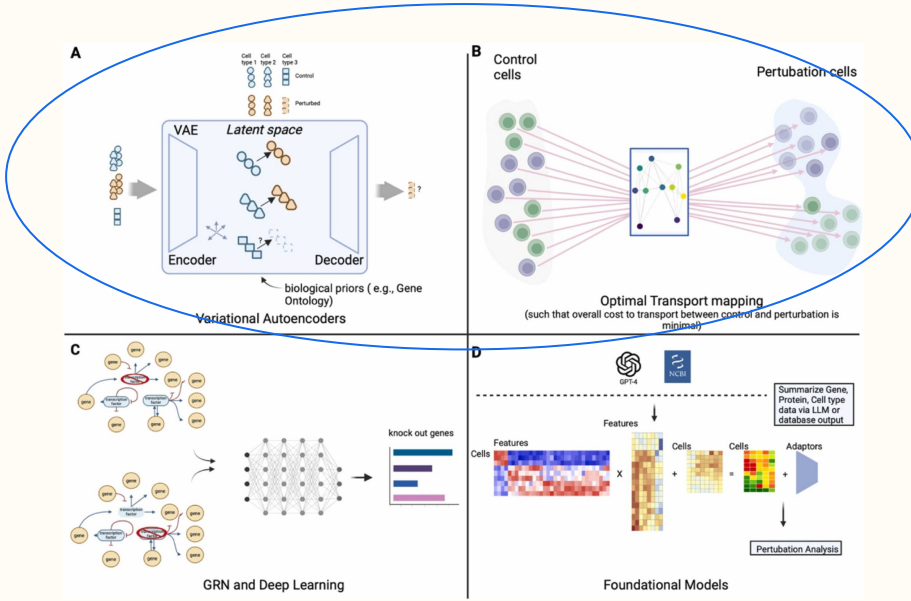
George I. Gavriilidis [a] · Vasileios Vasileiou [a,b] · Aspasia Orfanou [a] · Naveed Ishaque [c] · Fotis Psomopoulos [a]

Affiliations & Notes Article Info

https://doi.org/10.1016/j.csbj.2024.04.058

Focus of the current training event

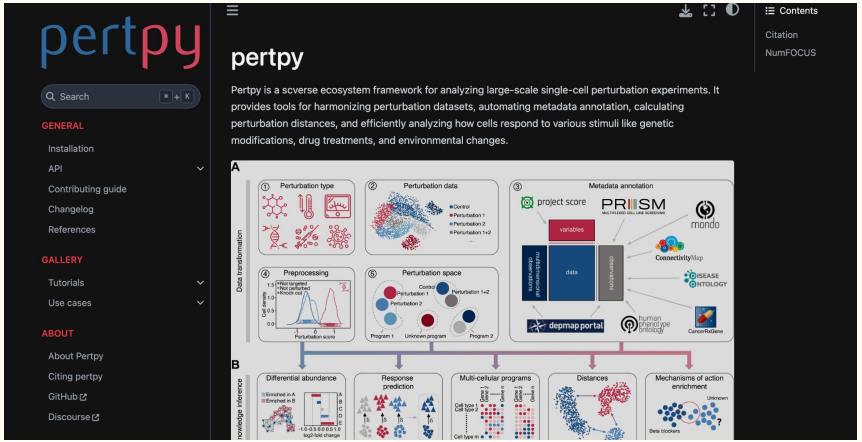




Why this training event?

Apart from the **`pertpy package`**, there is a profound scarcity of training material for these very intricate tools **(it only has ~7 tools though...)**

<u>Lack of</u> best practices, established benchmarks and standards impede the broader usage of these tools!



https://pertpy.readthedocs.io/en/latest/index.html

ISMB/ECCB 2025 Tutorial VT-8
Generative AI for Single-Cell Perturbation Modeling:
Theoretical and practical considerations

scGen: a landmark generative model for unseen perturbations

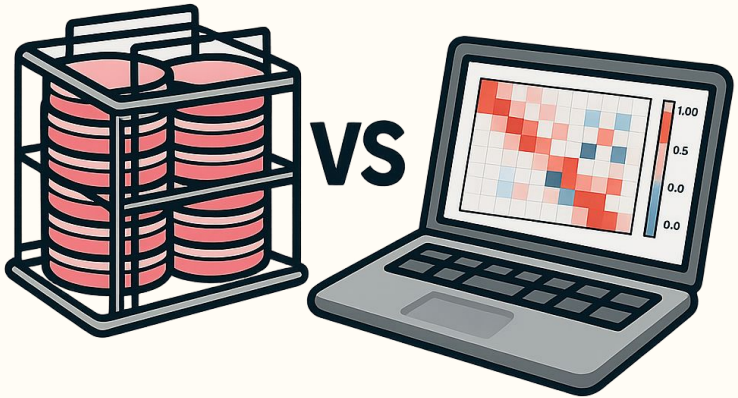
Konstantinos I. Giatras,
giatras@fleming.gr

HELLENIC REPUBLIC
National and Kapodistrian University of Athens
EST. 1837

"ALEXANDER FLEMING"
Biomedical Sciences Research Center

Why model perturbations *in silico*?

- 20 k genes × 400 cell types → wet lab screen impossible.
- Patient samples scarce & noisy.
- **In-silico models** rank experiments, suggest drug repurposing, personalise predictions.
- 2019: **scGen** shows DL can forecast unseen perturbations.

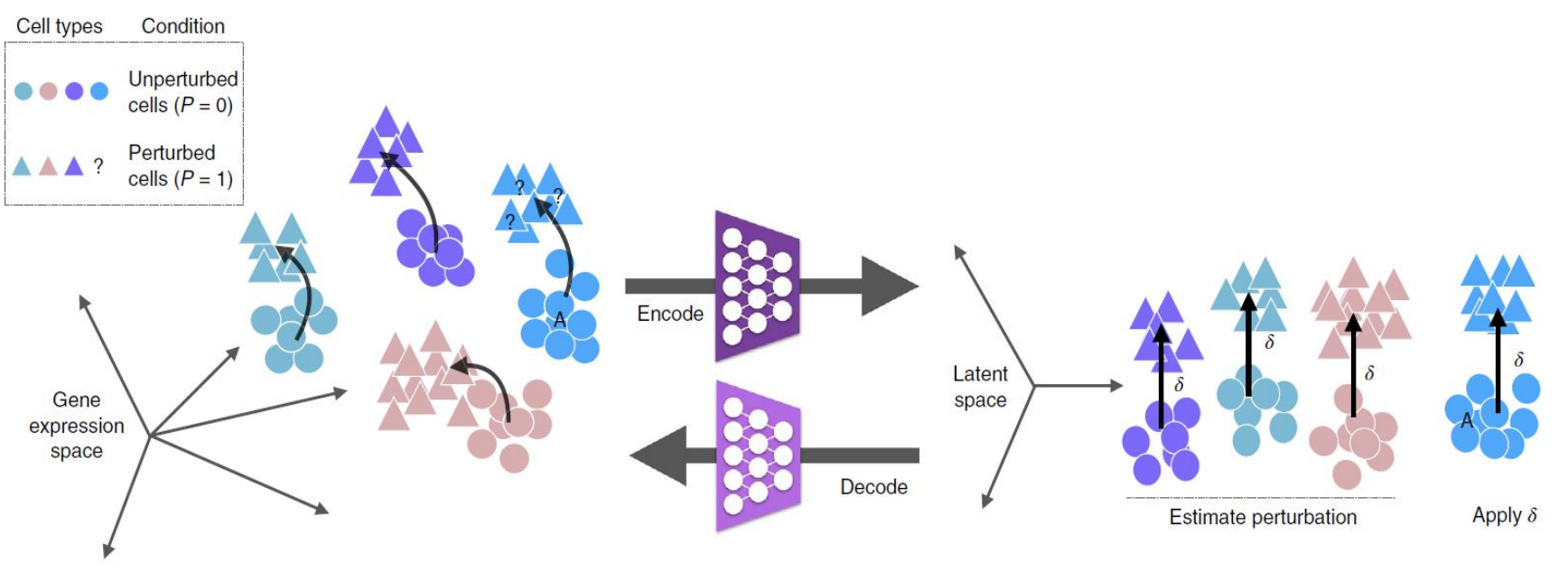


Decoding scGen-relevant jargon

Term	What it really means
Auto-encoder	Compresses a full gene profile, then rebuilds it.
β-VAE	Auto-encoder with a β weight that keeps features tidy and interpretable.
Latent space	Low-dimensional map where nearby points are transcriptionally similar cells.
ZINB decoder	Output layer tailored to zero-heavy single-cell RNA counts.

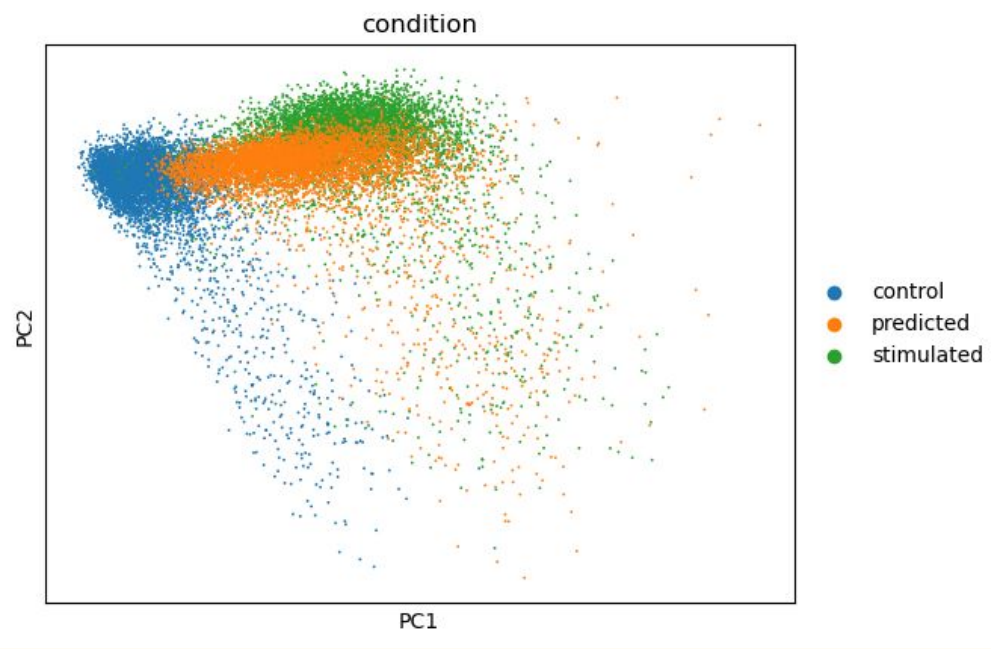
How scGen makes a prediction

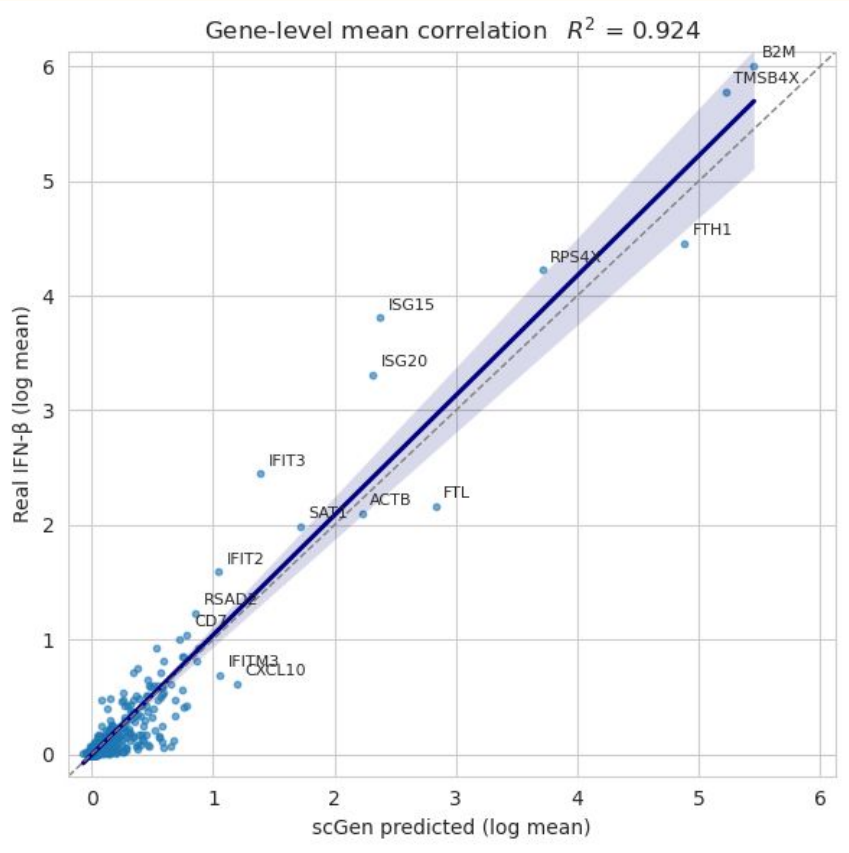
- β-variational auto-encoder (β-VAE).
- Reconstructs full gene expression profile – no pre-selected markers.
- Applicable to drugs, CRISPR KOs, cytokines, even dose series.



scGen output - PCA & mean-correlation snapshot

CD4 T controls vs IFN-β-stimulated example





Where scGen still shines …and why we call it a ‘legacy baseline’

✔ GPU-friendly & even runs on CPU	✘ No attention / pathway priors – ignores known gene–gene links
✔ δ vector is human-readable – easy to visualise & transfer	✘ One-size-fits-all δ – same shift for every cell hides heterogeneity
✔ Cross-species / cross-study transfer – δ often reusable across datasets	✘ Needs labelled cell types – can’t predict along unlabeled trajectories
✔ Fits neatly into the scvi-tools / Scanpy ecosystem	✘ Linear latent shift struggles with combos & dose series
✔ Stable VAE training with sensible defaults	✘ Outperformed by newer transformer / GNN models on some benchmarks

Key take-home messages

- **Simple idea, big payoff:** β-VAE + one δ vector shift.
- **Hardware-light baseline:** ~10 epochs on a laptop reach $R^2 \approx 0.9$ on the IFN-β test.
- **Interpretable & transferable:** the same δ explains biology, removes batch effects, even ports across species.
- **Mind the limits:** linear shift and labelled cell types miss pathway context, combos, low-expressed genes; transformers now cut the error.
- **Sets the stage:** everything you need for today's live notebook demo.

Next steps

- **Activate the tutorial environment** using conda: `conda activate scgen_tutorial`
- Start **Jupyter Lab** and open **scGen_Tutorial_ECCB2025.ipynb**
- **Hands-on demo**: train scGen on the local PBMC dataset for 10 quick epochs, predict the IFN-β response of held-out CD4 T cells, and explore the results live with PCA/UMAP plots, gene-wise R^2, and various insightful metrics.
- We will go through the entire notebook **together** (message us if you encounter any technical difficulties)
- All material CC-BY (open license)

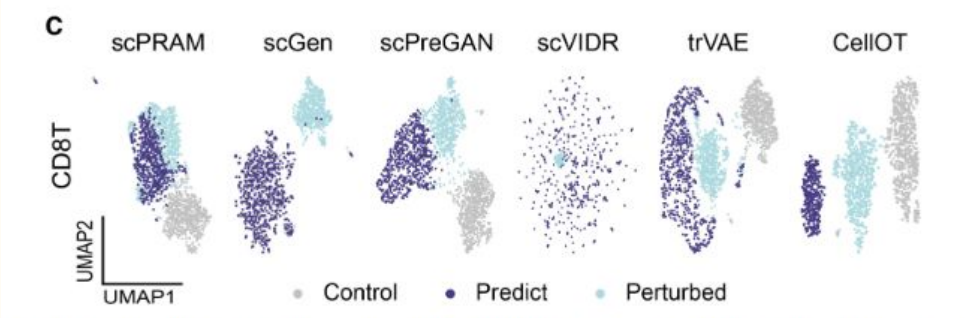
ISMB/ECCB 2025 Tutorial VT-8
Generative AI for Single-Cell Perturbation Modeling:
Theoretical and practical considerations

scPRAM: an attention-based take on perturbation modelling

Sabrina Jagot,
sabrina.jagot@univ-lyon1.fr

State of the art when scPRAM arrives

- Existing predictors mostly **average over cells or over cell types**, losing cell-specific heterogeneity.
- Existing predictors efficiency was subject a lot with noise (sparcity) levels in datasets.
- 2024: **scPRAM** change the game with their goal ⇒infer each cell's full gene-expression response—even for **unseen cell-types, species or patients**.



Jiang et al., 2024

Perturbation response based on attention mechanism

1. Variational Auto-Encoder (VAE)

$$X \rightarrow Z$$

2. Sinkhorn Optimal Transport

$$M = \text{Sinkhorn}(Z^{\text{ctrl}}, Z^{\text{ptb}})$$

3. Cell-specific perturbation vectors

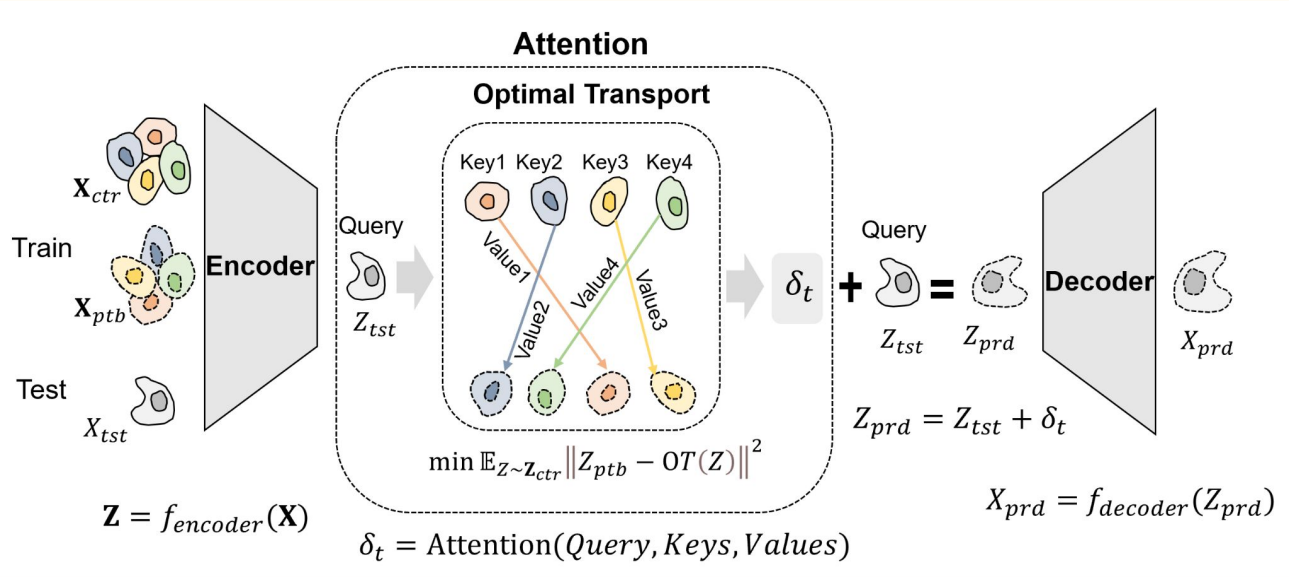
$$\delta_i = Z_i^{\text{ptb}} - Z_i^{\text{ctrl}}$$

4. Attention mechanism

$$\hat{\delta}_t = \sum w_i \delta_i$$

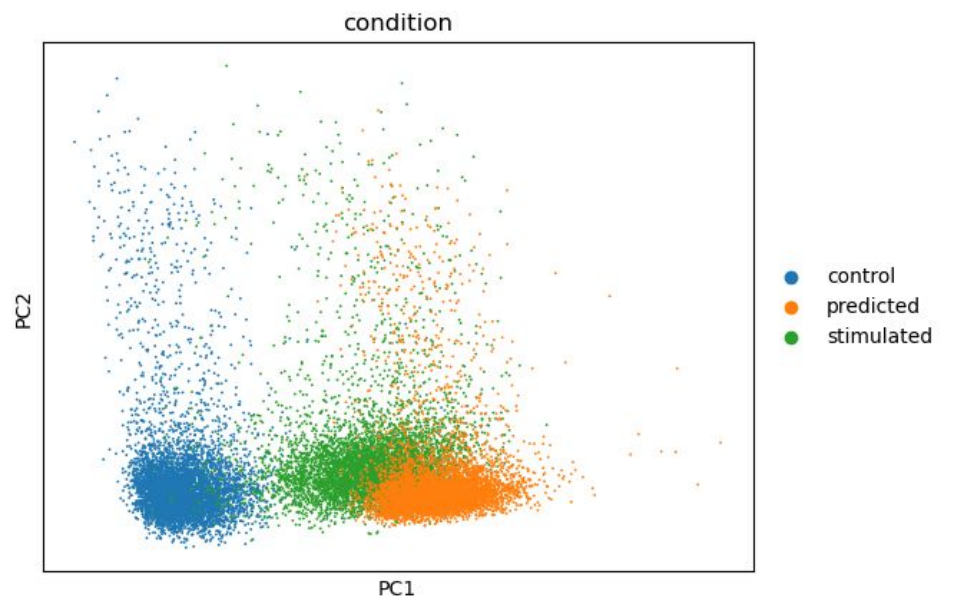
5. Decoder

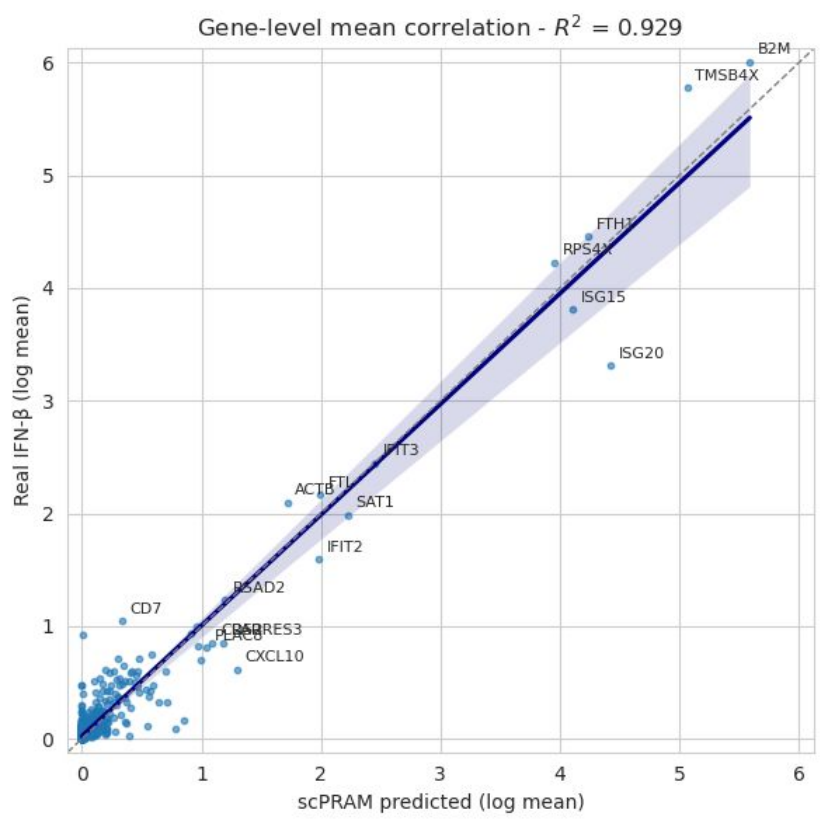
$$\hat{X}^{\text{ptb}} = \text{Dec}(z_t + \hat{\delta}_t)$$



A cell-type prediction visualization example

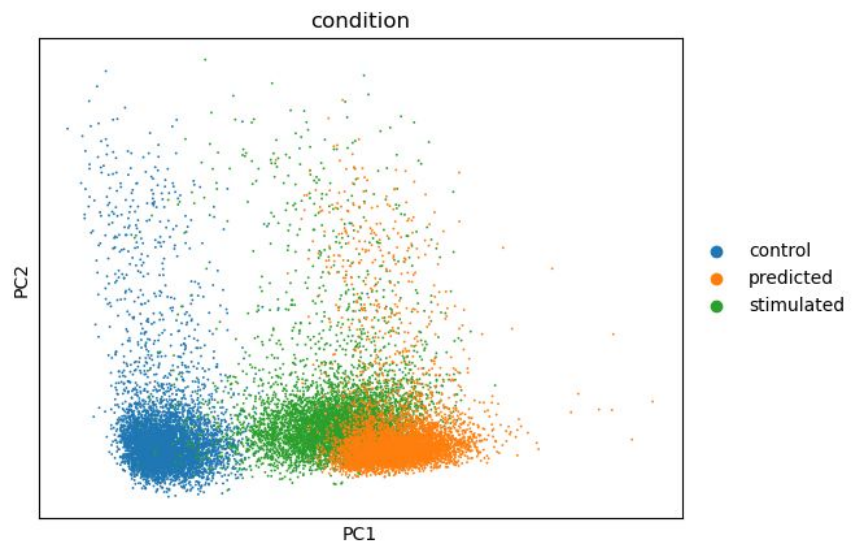
CD4 T controls vs IFN-β-stimulated example

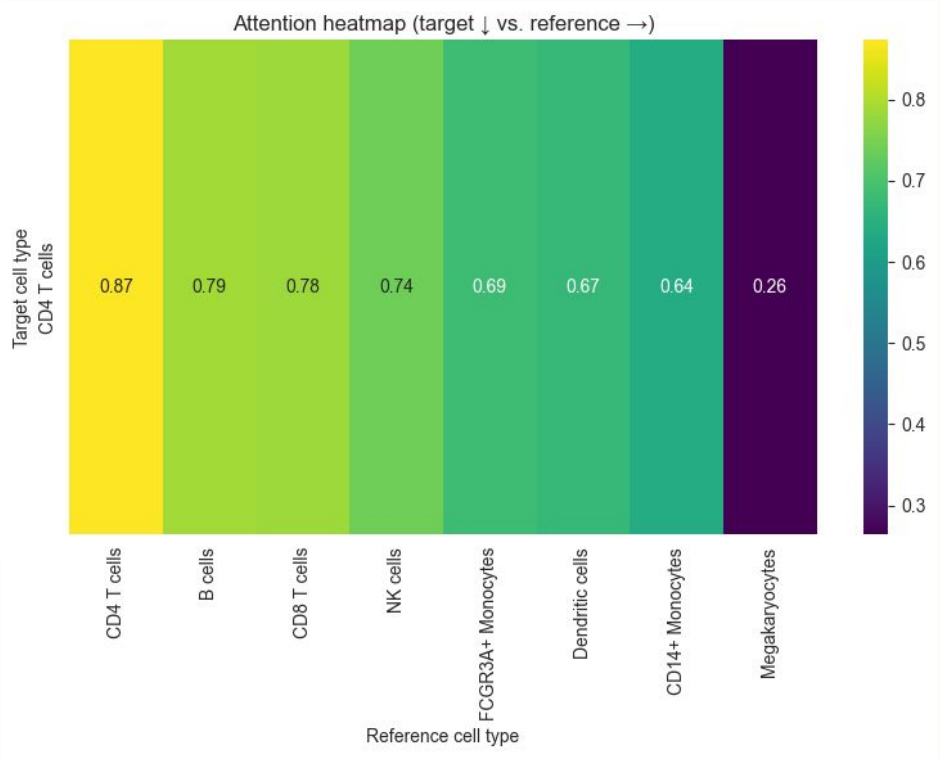




A cell-type prediction visualization exemple

CD4 T controls vs IFN-β-stimulated: attention score, a big deal ?





scPRAM's strengths and weaknesses

✔ GPU-friendly & even runs on CPU	✘ Cross-species / cross-study transfer – no documentation on how to use scPRAM for that
✔ Attention mechanism – specific to each cell, where the weights reflect its actual proximity in latent space	✘ Transparent latent algebra – less easier to interpret **δ** vectors
✔ Heterogeneity over averages – adds the specific **δ** to every cell	✘ Addition of parameters to adjust modelling - can be long time efforts for making a good prediction
✔ No needs labelled cell types	✘ Linear latent shift struggles with combos & dose series

Key take-home messages

- **Attention-driven perturbation:** scPRAM's key innovation = per-cell attention mechanism.
- **Robust within-dataset performance:** high R^2 and low energy-distance metrics.
- **Transparent cross-study transfer:** Our workflow demonstrates that scPRAM can predict perturbations in a new cohort—providing a reproducible protocol missing from the literature.
- **Integration & evaluation pipeline:** Successful extrapolation depends on robust batch-correction, followed by rigorous benchmarking (R^2, MSE, energy/KDE distances, DEG overlap, attention heatmaps) to validate transfer fidelity.
- **Sets the stage:** everything you need for today's live notebook demo.

Next steps

- **Activate the tutorial environment** using conda: `conda activate scpram_tutorial`
- Start **Jupyter Lab** and open **scPRAM_Tutorial_ECCB2025.ipynb**
- **Hands-on demo**: train scPRAM on the local PBMC dataset for 10 quick epochs, predict the IFN-β response of held-out CD4 T cells, and explore the results live with PCA/UMAP plots, gene-wise R^2, and various insightful metrics.
- Explore the **attention mechanism** and different models provide by scPRAM
- We will go through the entire notebook **together** (message us if you encounter any technical difficulties)
- All material CC-BY (open license)

ISMB/ECCB 2025 Tutorial VT-8
Generative AI for Single-Cell Perturbation Modeling:
Theoretical and practical considerations

Benchmarking perturbation modelling tools

Alejandro Madrid Valiente,
alejandro.madrid@bsc.es

Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

Why benchmark single-cell perturbation tools?

Diverse tools, divergent assumptions

Ensure reproducibility and robustness. Helps to identify tools that perform consistently across datasets, platforms and perturbation types

Evaluate under realistic conditions. A meaningful benchmark should simulate ''data realism''.

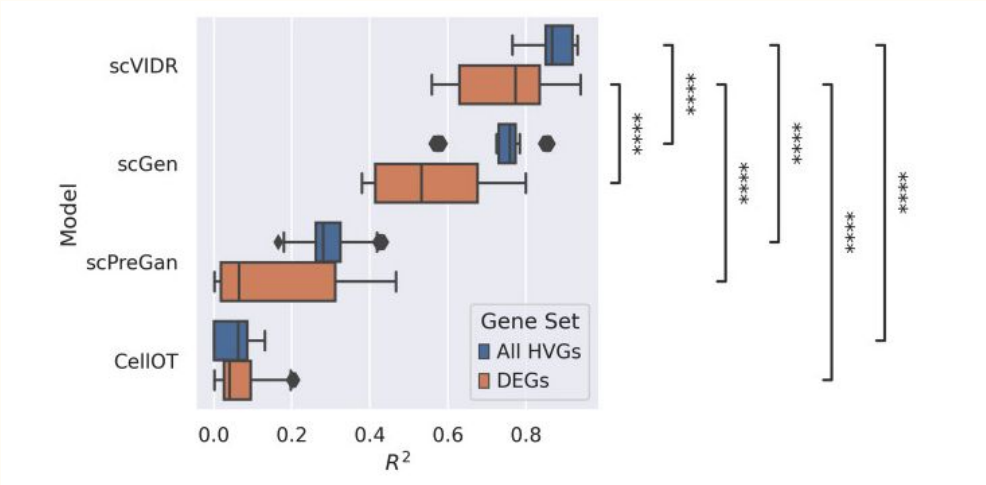
Guide method selection and development. Helps revealing which tools perform better in specific tasks.

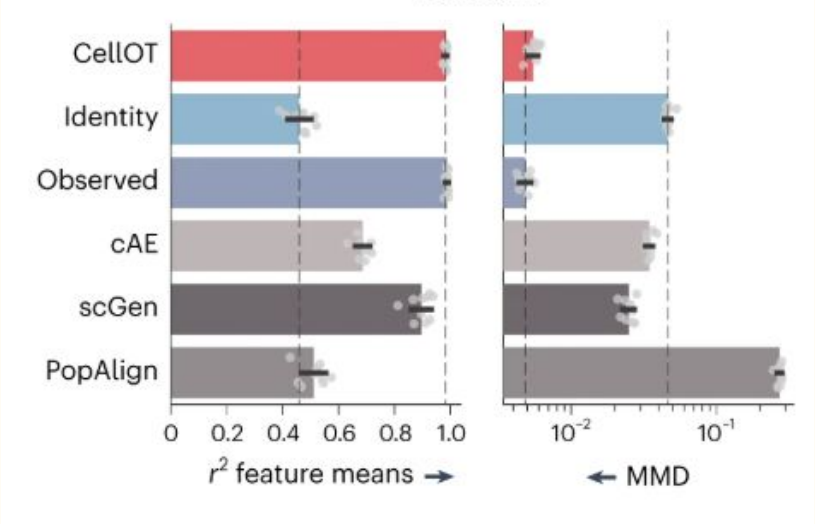
Trustworthy biological interpretation

The curious case of CellOT

Is CellOT better than scGEN or not??

Different datasets-metrics-standards; Lack of best practices!



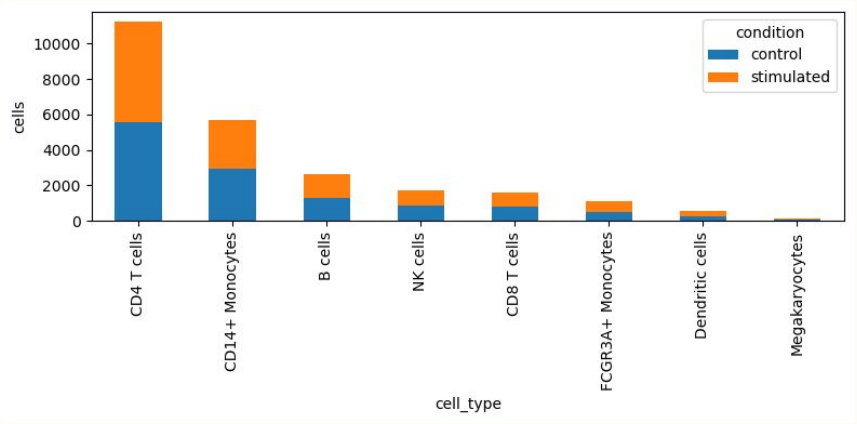


https://doi.org/10.1016/j.patter.2023.100817

https://www.nature.com/articles/s41592-023-01969-x#change-history

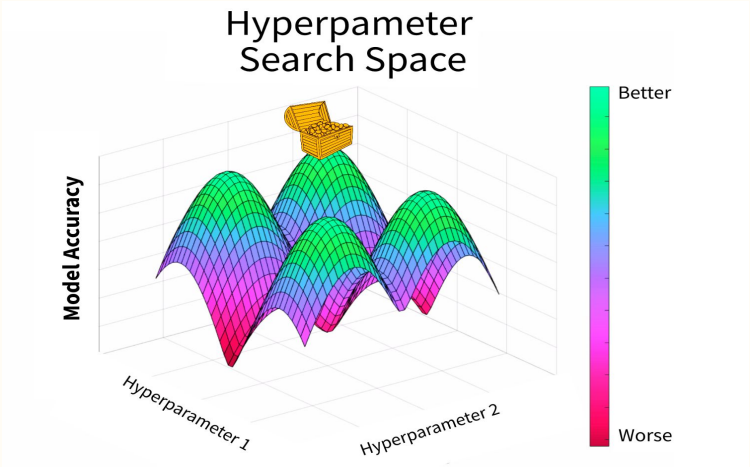
Benchmark Design.

Part 1. Dataset proportion



Which tool scGEN or scPRAM will perform better in an **unbalanced** dataset compared with the original one ?

Part 2. Hyperparameter tuning



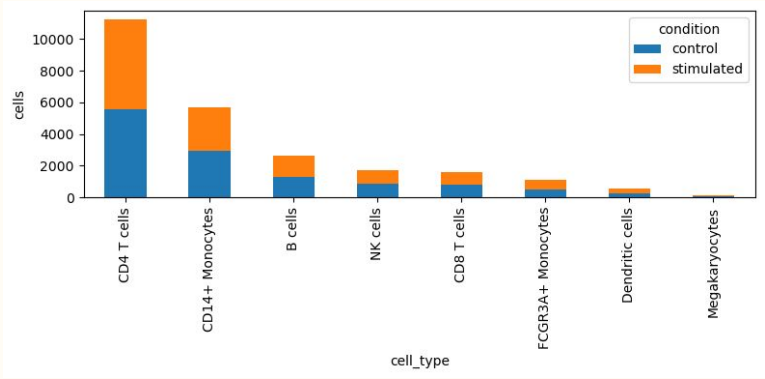
Each tool has several hyperparameters. Which is the **best combination**? Which one gives more **variability** to the results?

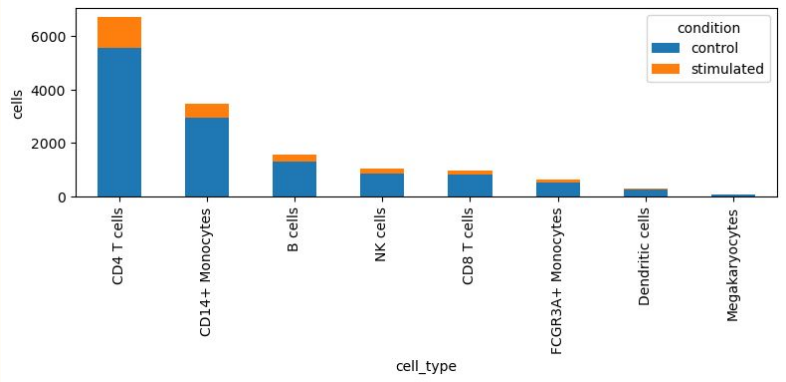
Part 1. Dataset proportion

We generated an **unbalanced** dataset which is closer to the reality.

We should check differences between the **DEGs** apart from the normal metrics.

Data realism matters as much as **model complexity**.





Part 2. Hyperparameter tuning

Hyperparameters are the knobs that control model architecture and training, they are ***not learned*** during training but set beforehand.

The choice of hyperparameters can significantly affect model performance, especially in high-dimensional and sparse single-cell data.

We will use **Optuna** which uses **Bayesian optimization** and pruning to find the best hyperparameters faster than grid or random search.

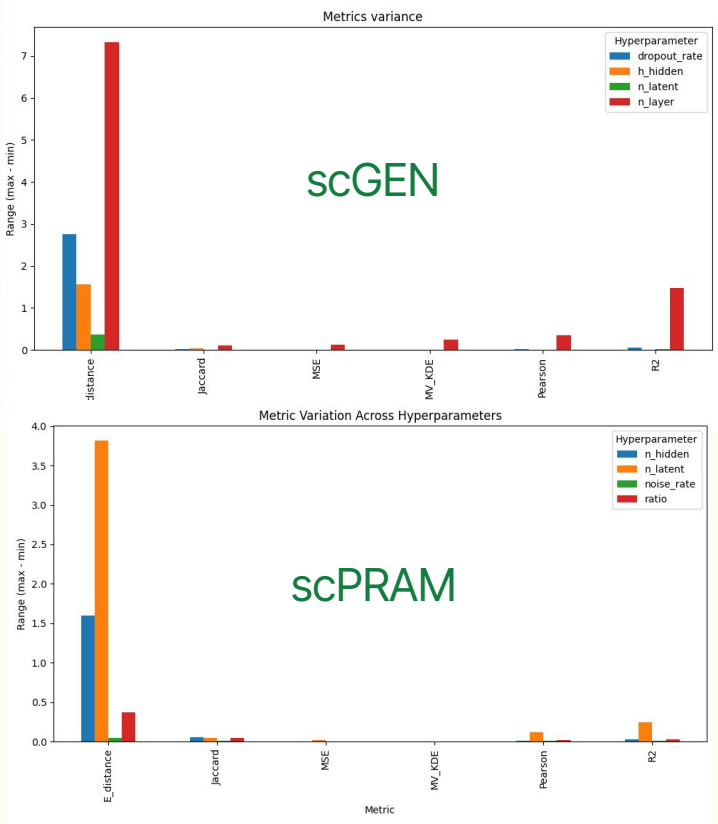
HYPERPARAMETER	WHAT CONTROLS ?	TOO HIGH VALUES	TOO LOW VALUES
N_LAYERS	Depth of neural network	longer training, risk of overfitting	limited model capacity
N_LATENT	Dimensionality of the compressed representation	overfitting, noisy clustering	important biological variation might be lost
N_HIDDEN	Width of hidden layers	overfitting or slow training	poor latent space structure
DROPOUT_RATE	Fraction of neurons randomly "dropped" during training	underfitting	overfitting
NOISE_RATE	Injects random noise into input data	blurs real data	less robustness, overfitting risk
RATIO	how many neighbours feed the attention mechanism	over-smooth the delta	unstable

Part 2. Hyperparameter tuning

Effective tuning of hyperparameters is essential to achieve **optimal model performance**, ensuring accurate **generalization** and **stable behavior** across perturbation settings

In **scGEN** *n_layers* and *dropout rate* seem to be the most sensitive to values changes

In **scPRAM** *n_latent* and *n_hidden* seem to be the most sensitive to values changes



Other benchmarking analysis

Cross-study extrapolation. Which tool performs better on different batches, technologies or donors

Cell type specificity. Do methods generalize across diverse cell types, or are they biased toward abundant or well-annotated populations?

Key take-home messages

- **Handling unbalanced datasets is crucial:** Data realism matters as much as model complexity
- **Hyperparameter optimization significantly impacts results:** systematic tuning is essential to avoid overfitting or underfitting
- **Balanced evaluation metrics provide a fuller picture:** Metrics sensitive to imbalance (E-distance) complement global fit metrics (R^2, MSE)

Next steps

- Start **Jupyter Lab** and open **Benchmarking_Tutorial_ECCB2025_Part1.ipynb**
- **Hands-on demo**: you will compare results depending on the balanced/unbalanced dataset. You will also compare the results depending on the hyperparameter combination
- We will go through the entire notebook **together** (message us if you encounter any technical difficulties)
- All material CC-BY (open license)

Take away messages

- EU BH 2024 Perturb-Bench event is ongoing
  https://github.com/BiodataAnalysisGroup/BioHackathon

- Our vision is to make a **FAIR decentralized platform for benchmarking,** through NextFlow, leveraging the ELIXIR infrastructure like WorkflowHub and RO-Crates

- Join us in November in the EU BH 2025 in Berlin for our new endeavor on foundational/LLM-like models on single-cell omics

Acknowledgements

Marina Esteban Medina

Fotis Psomopoulos

Naveed Ishaque

Ali Kerim Secener

Liya Zaygerman

Rosario Astaburuaga-García

Sven Klumpe

Aspa Orfanou

Vasileios Vasileiou