Combining deep learning and modeling for time-series single-cell RNA-sequencing data

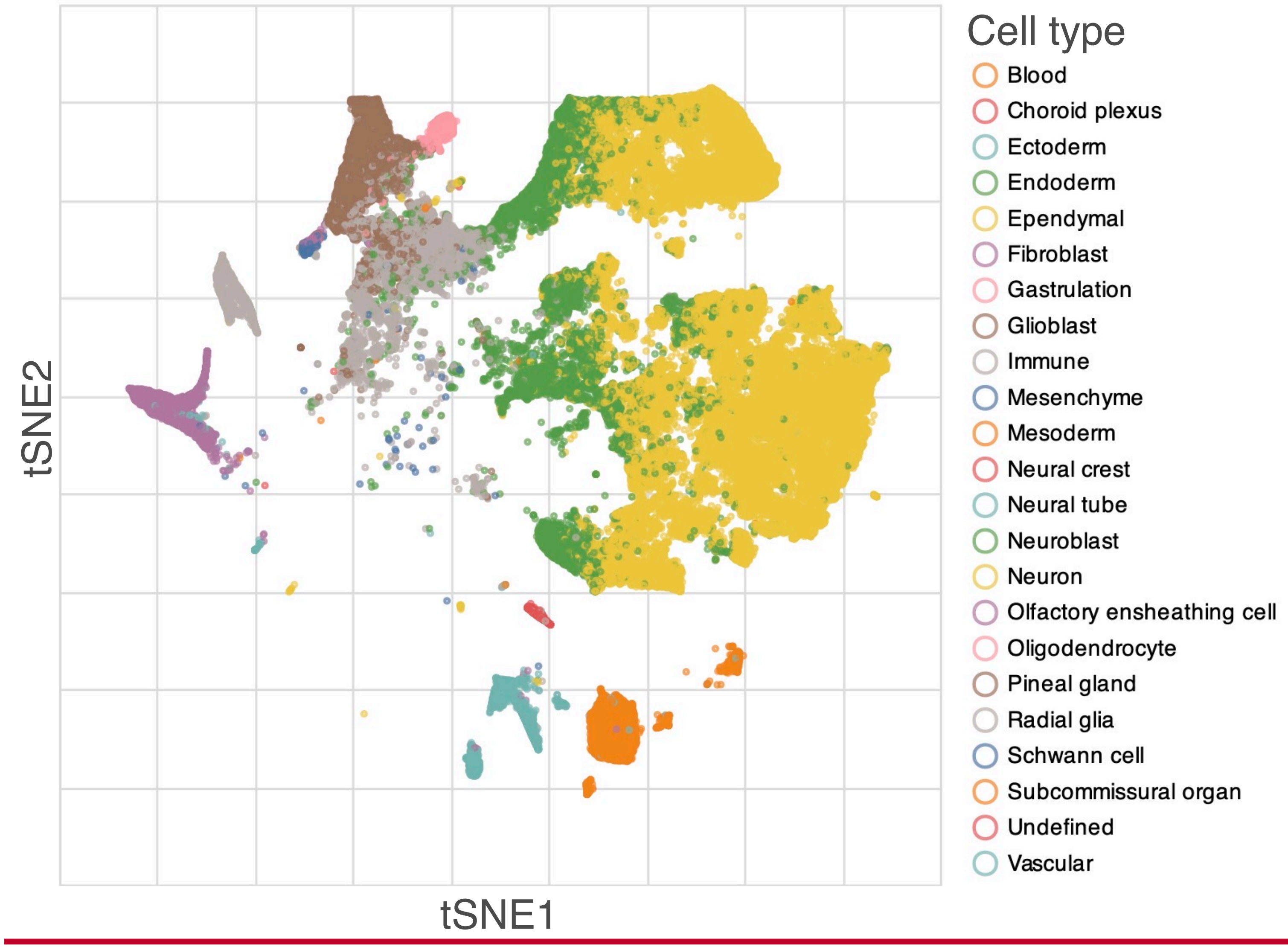
Maren Hackenberg, Harald Binder

Institute of Medical Biometry and Statistics (IMBI), Faculty of Medicine and Medical Center – University of Freiburg

Bioinformatics Club – April 11, 2022

How do you typically look at single-cell RNA-seq data?

t-stochastic neighborhood embedding (tSNE) for dimension reduction

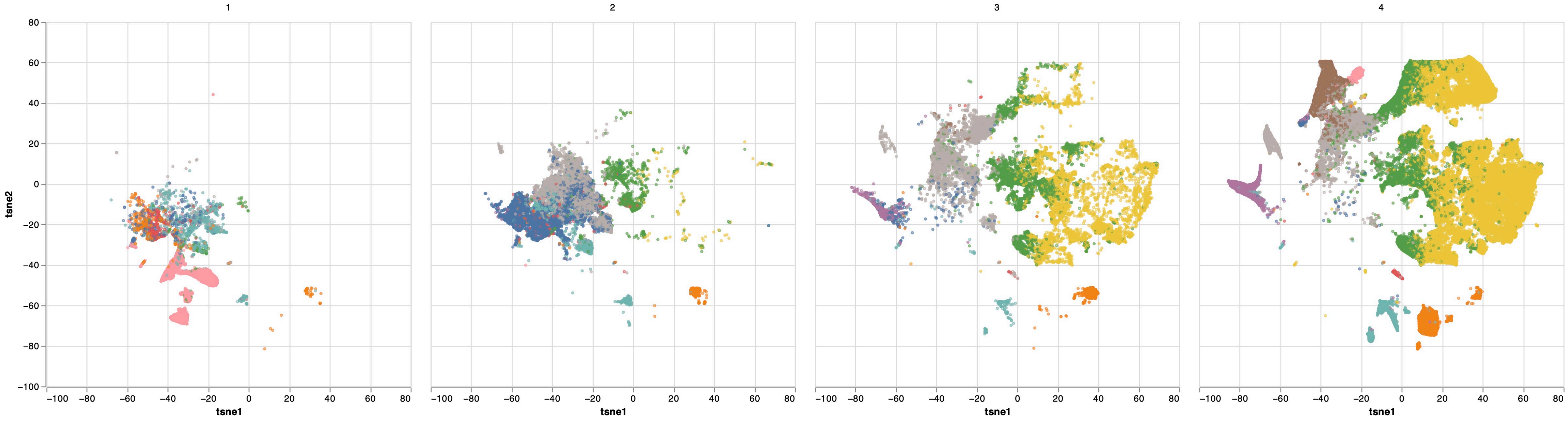


Data from La Manno et al. (2021) Molecular architecture of the developing mouse brain. *Nature* 596, 92-96.

How do you typically look at single-cell RNA-seq data?

t-stochastic neighborhood embedding (tSNE) for dimension reduction

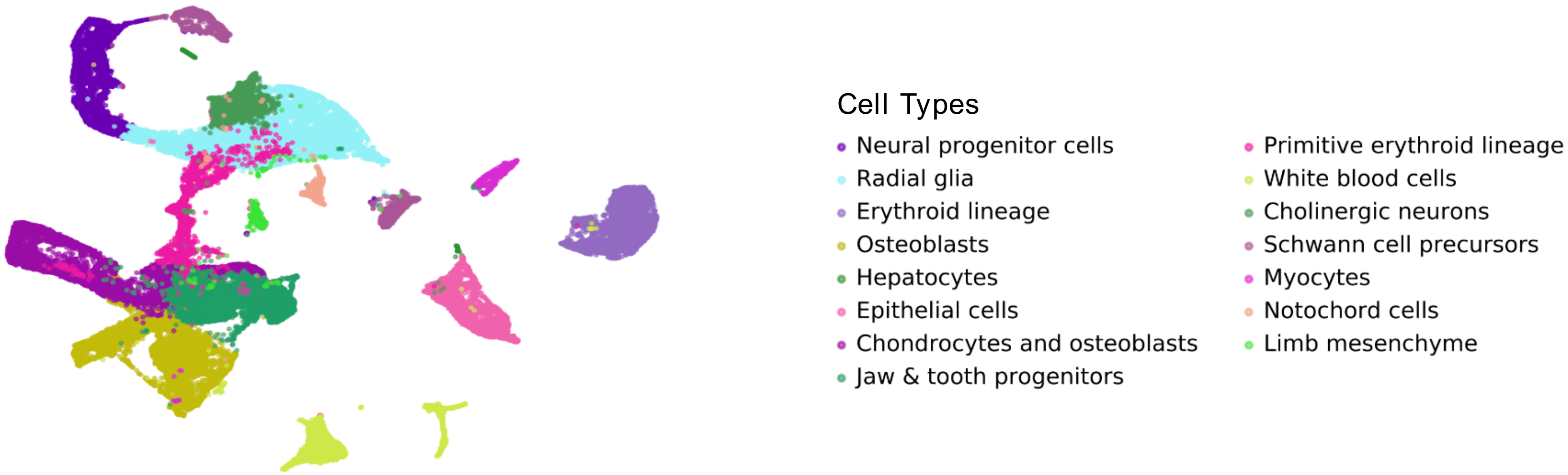
Developmental stage

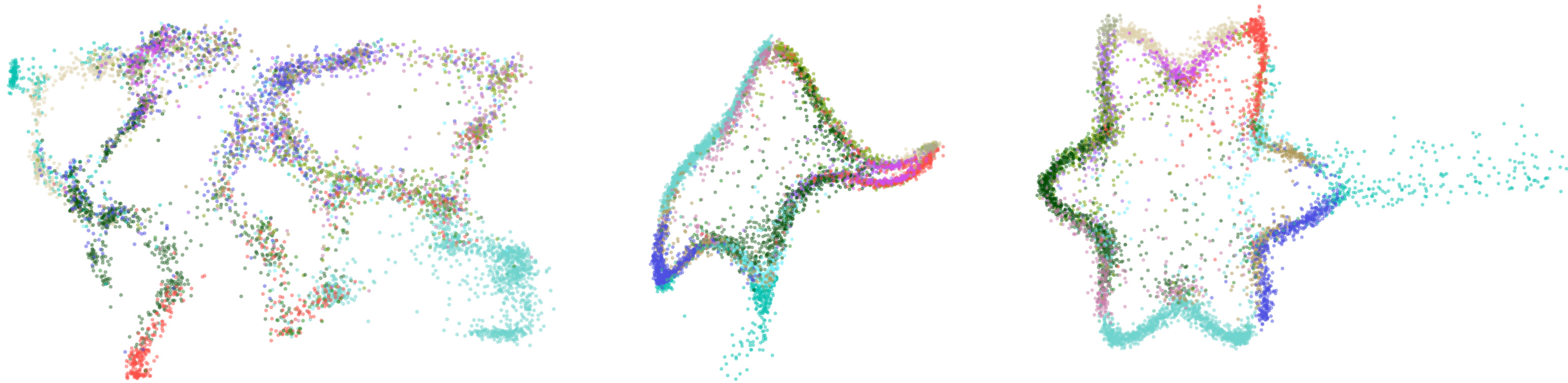


Data from La Manno et al. (2021) Molecular architecture of the developing mouse brain. *Nature* 596, 92-96.

Dimension reduction can be misleading

Data: ex utero mouse embryo E8.5



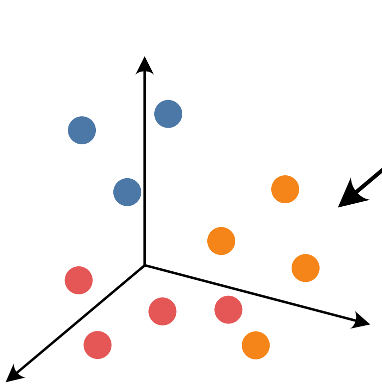


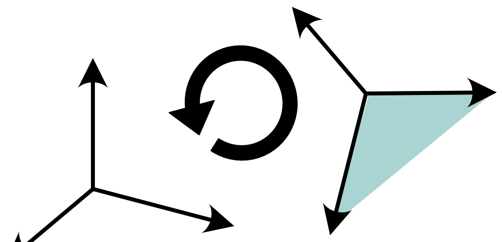
Figures taken from Chari et al. (2021) The specious art of single-cell genomics. *bioRxiv preprint,* doi*:* 10.1101/2021.08.25.457696.

UNIVERSITÄTS KLINIKUM FREIBURG

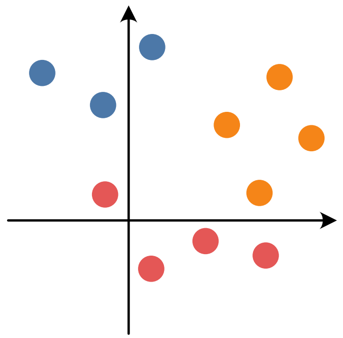
Comparing popular dimension reduction methods

PCA

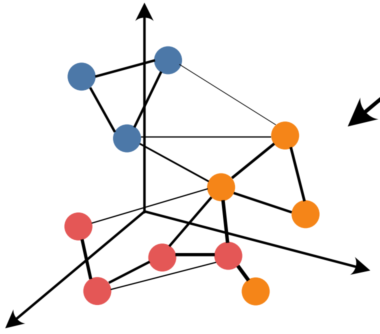


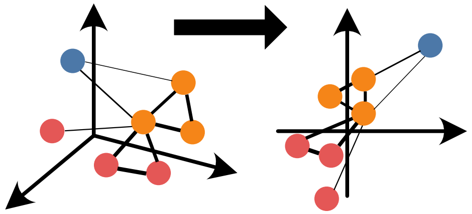


rotate and project to axes of greatest variation

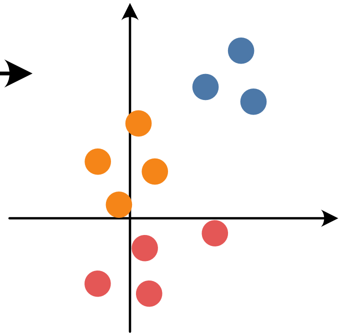


UMAP

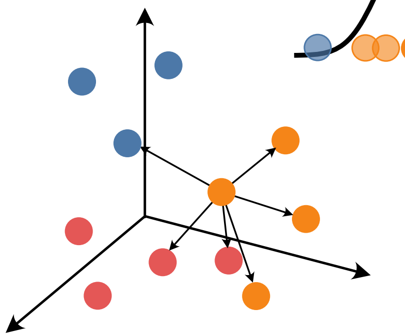


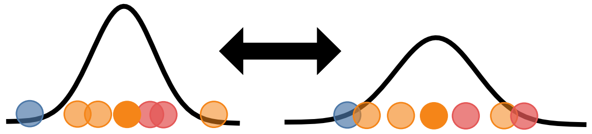


preserve graph structure

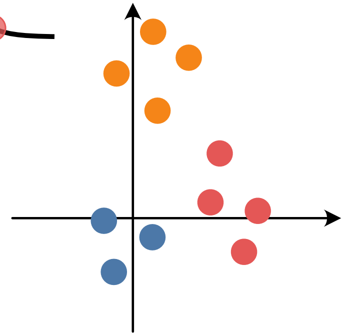


tSNE

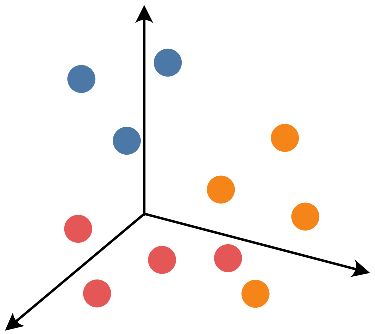


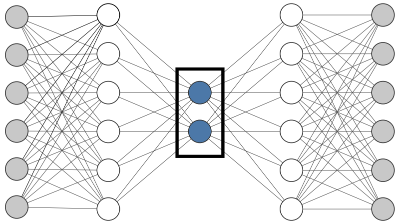


preserve distributions of distances to nearest neighbors

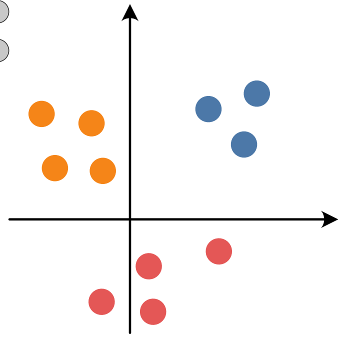


VAE

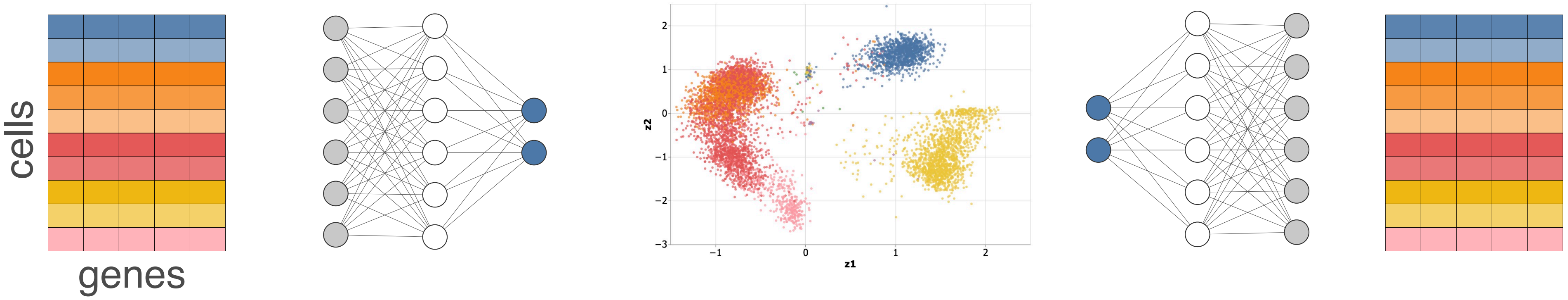




use neural-network based Variational Autoencoder

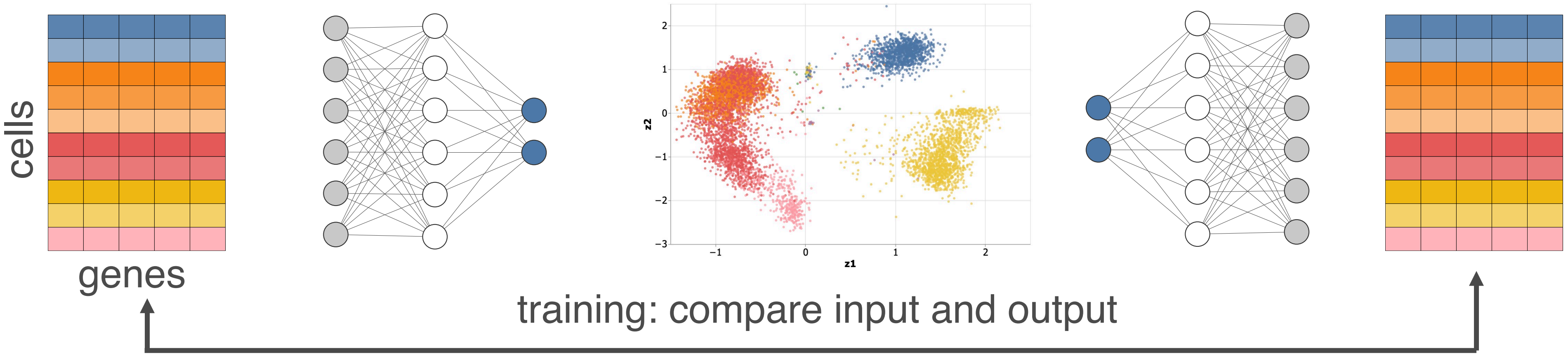


Creating synthetic temporal patterns in scRNA-seq data



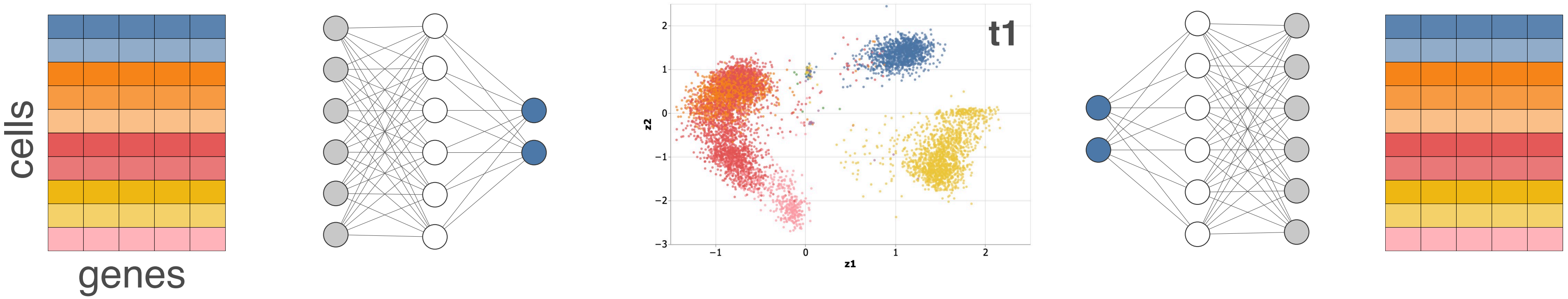
1. Use a VAE to learn a low-dimension representation of a snapshot scRNA-seq dataset

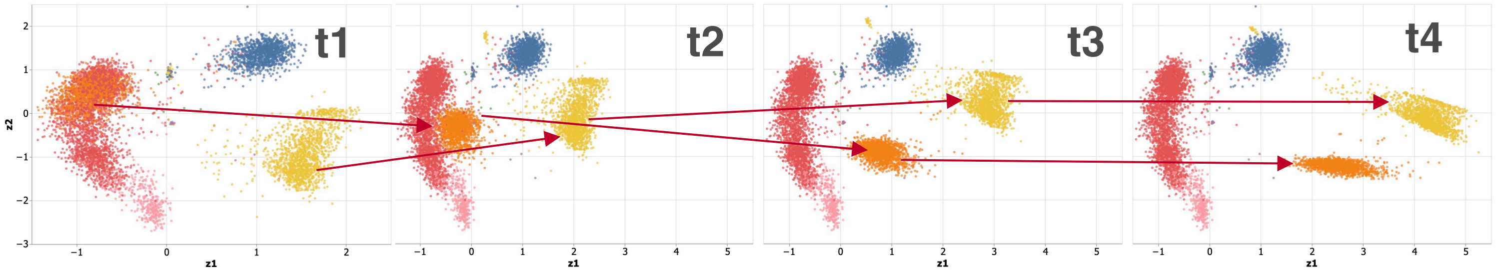
Creating synthetic temporal patterns in scRNA-seq data



1. Use a VAE to learn a low-dimension representation of a snapshot scRNA-seq dataset

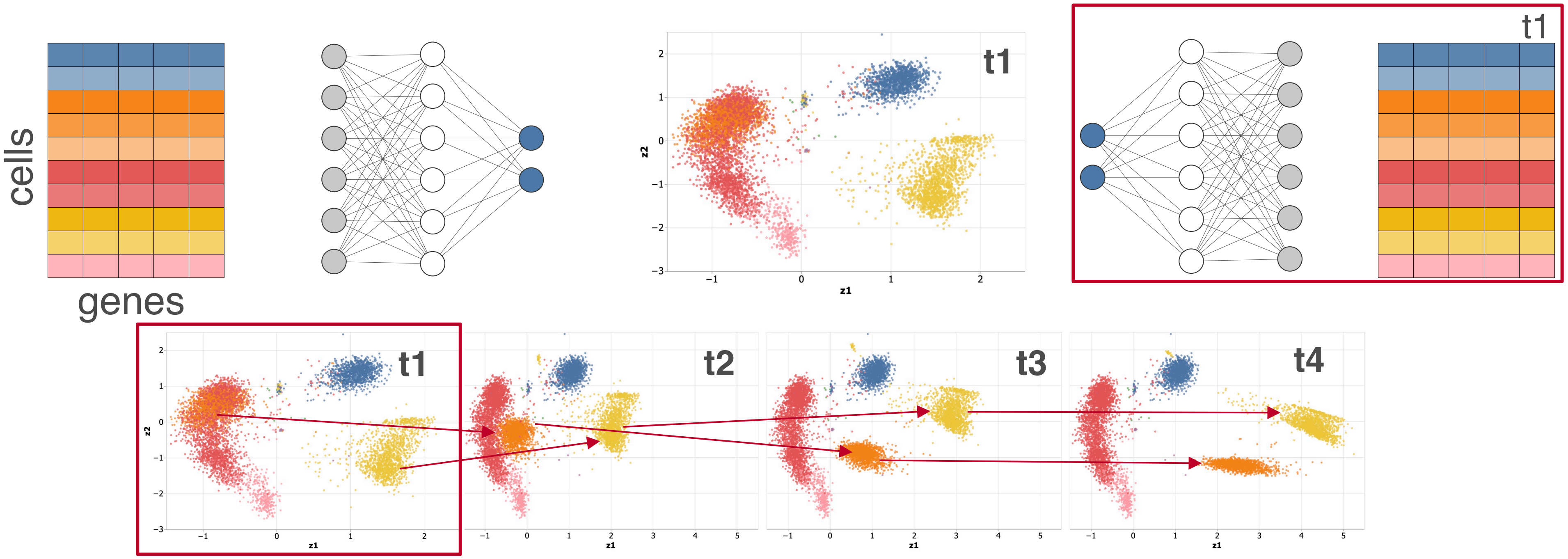
Creating synthetic temporal patterns in scRNA-seq data





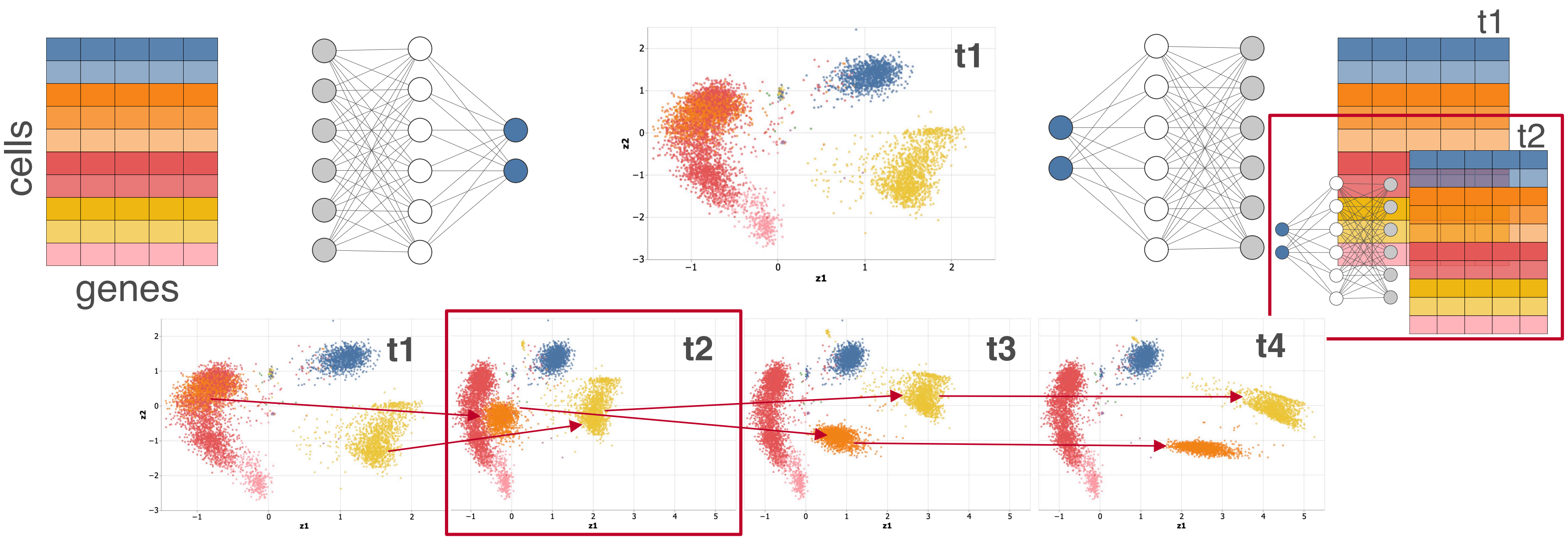
1. Use a VAE to learn a low-dimension representation of a snapshot scRNA-seq dataset
2. In the latent representation, induce a synthetic temporal pattern by vector field transformations

Creating synthetic temporal patterns in scRNA-seq data



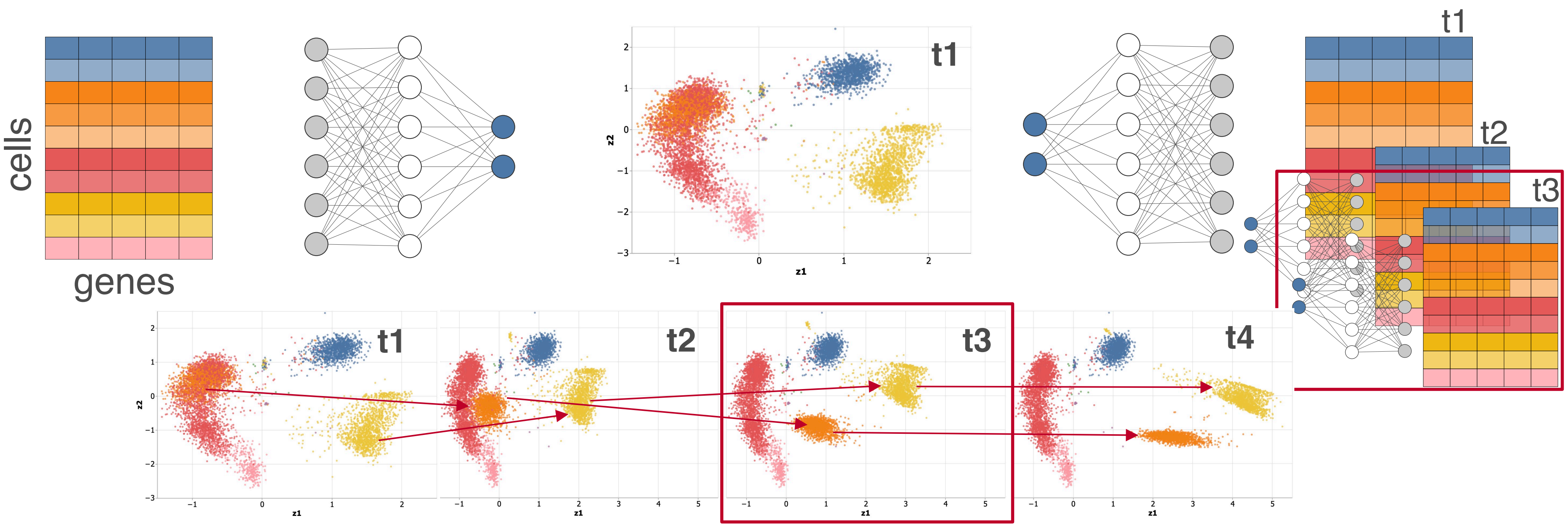
1. Use a VAE to learn a low-dimension representation of a snapshot scRNA-seq dataset
2. In the latent representation, induce a synthetic temporal pattern by vector field transformations
3. Use the VAE decoder to generate corresponding high-dimensional data for each time point

Creating synthetic temporal patterns in scRNA-seq data



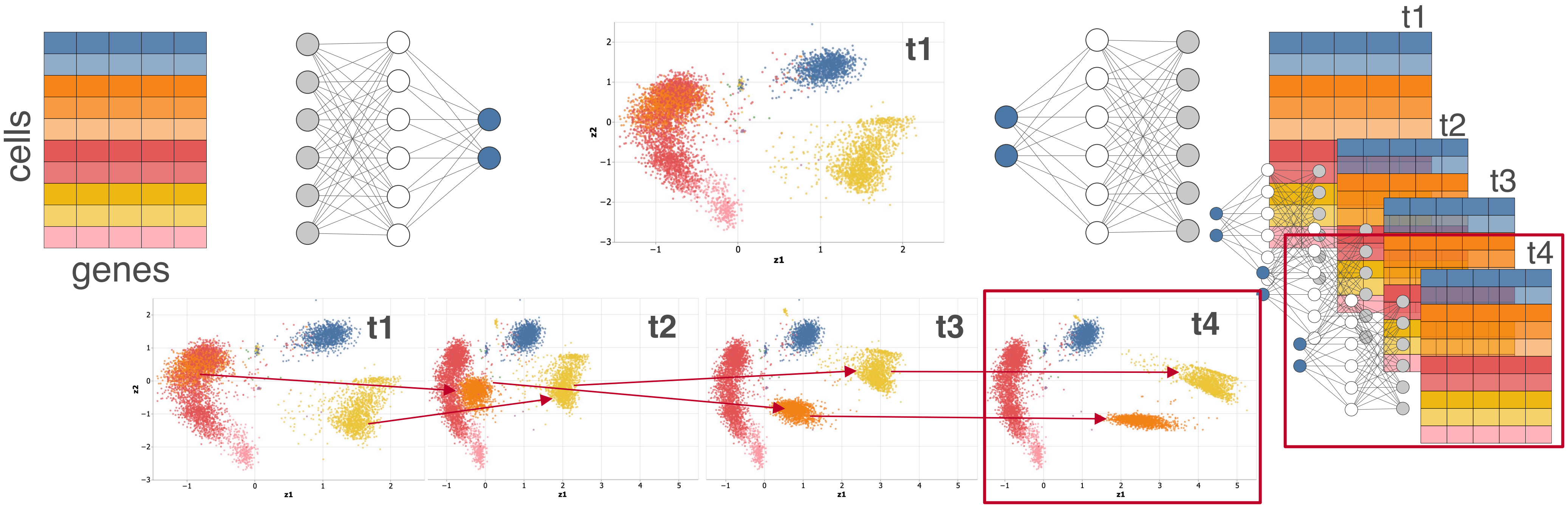
1. Use a VAE to learn a low-dimension representation of a snapshot scRNA-seq dataset
2. In the latent representation, induce a synthetic temporal pattern by vector field transformations
3. Use the VAE decoder to generate corresponding high-dimensional data for each time point

Creating synthetic temporal patterns in scRNA-seq data



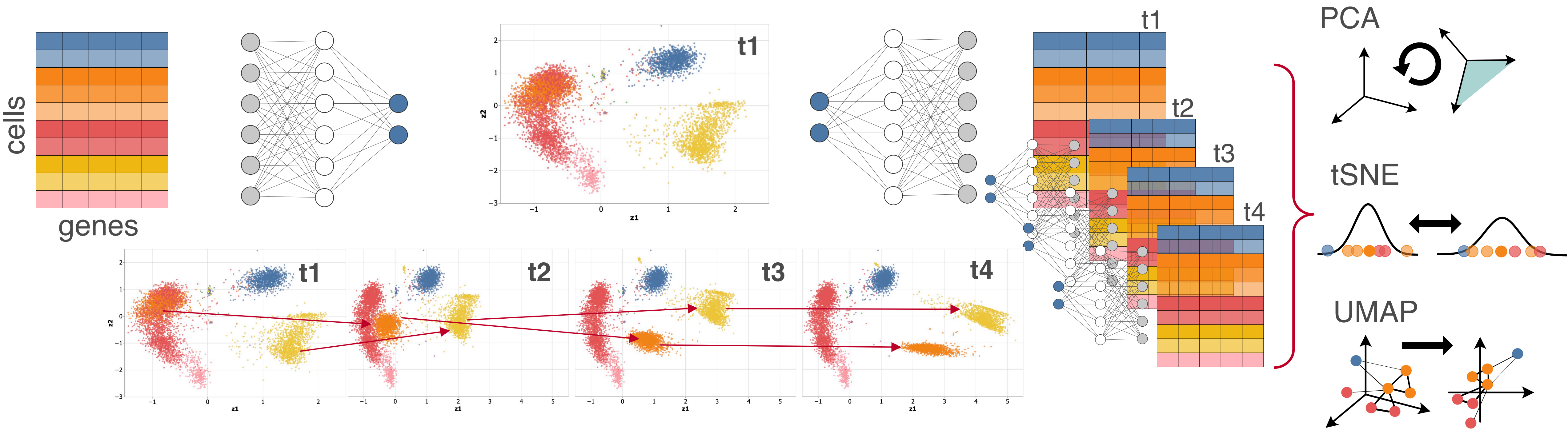
1. Use a VAE to learn a low-dimension representation of a snapshot scRNA-seq dataset
2. In the latent representation, induce a synthetic temporal pattern by vector field transformations
3. Use the VAE decoder to generate corresponding high-dimensional data for each time point

Creating synthetic temporal patterns in scRNA-seq data



1. Use a VAE to learn a low-dimension representation of a snapshot scRNA-seq dataset
2. In the latent representation, induce a synthetic temporal pattern by vector field transformations
3. Use the VAE decoder to generate corresponding high-dimensional data for each time point

Creating synthetic temporal patterns in scRNA-seq data

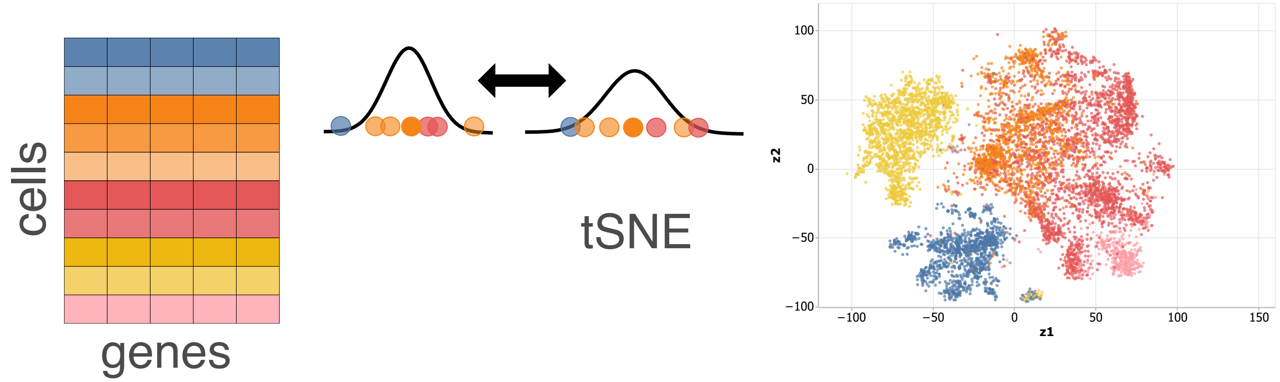


1. Use a VAE to learn a low-dimension representation of a snapshot scRNA-seq dataset
2. In the latent representation, induce a synthetic temporal pattern by vector field transformations
3. Use the VAE decoder to generate corresponding high-dimensional data for each time point
4. Apply PCA, tSNE and UMAP on the synthetic high-dimensional time-series data

UNIVERSITÄTS KLINIKUM FREIBURG

Inducing temporal patterns in tSNE-space

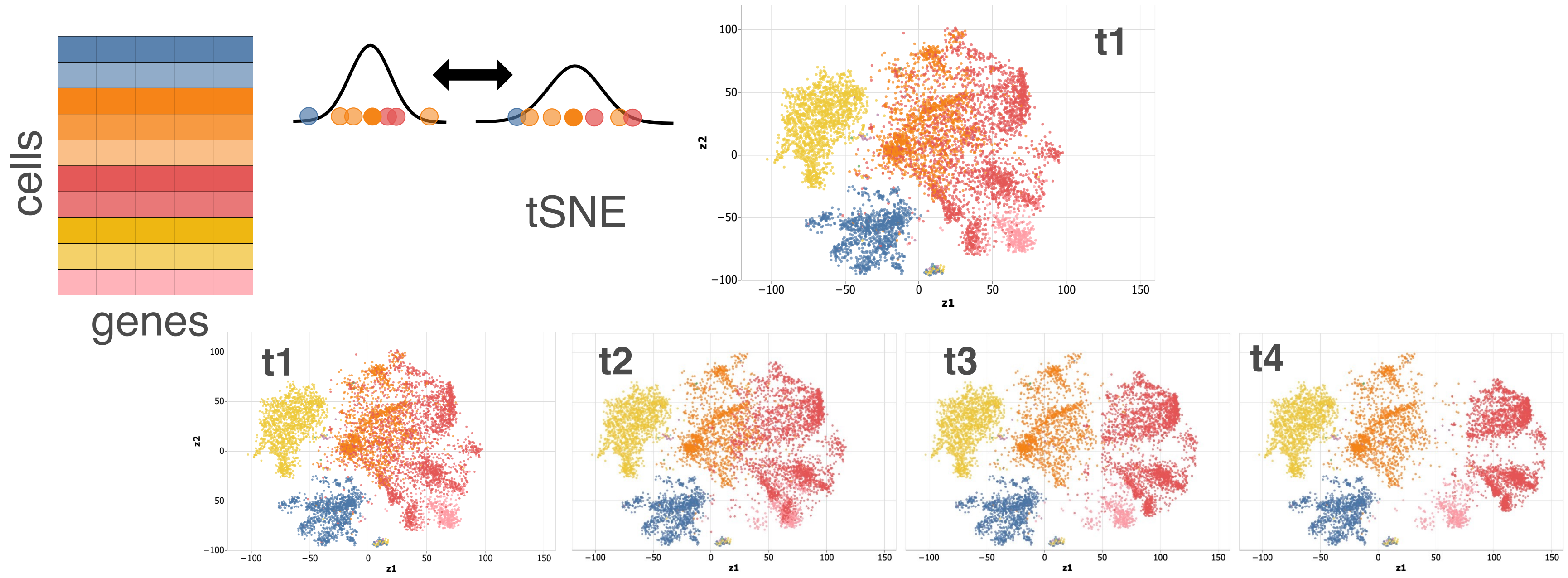
How to generate high-dimensional data from the low-dimensional time series?



1. Apply tSNE (or PCA/UMAP) to a snapshot scRNA-seq dataset

Inducing temporal patterns in tSNE-space

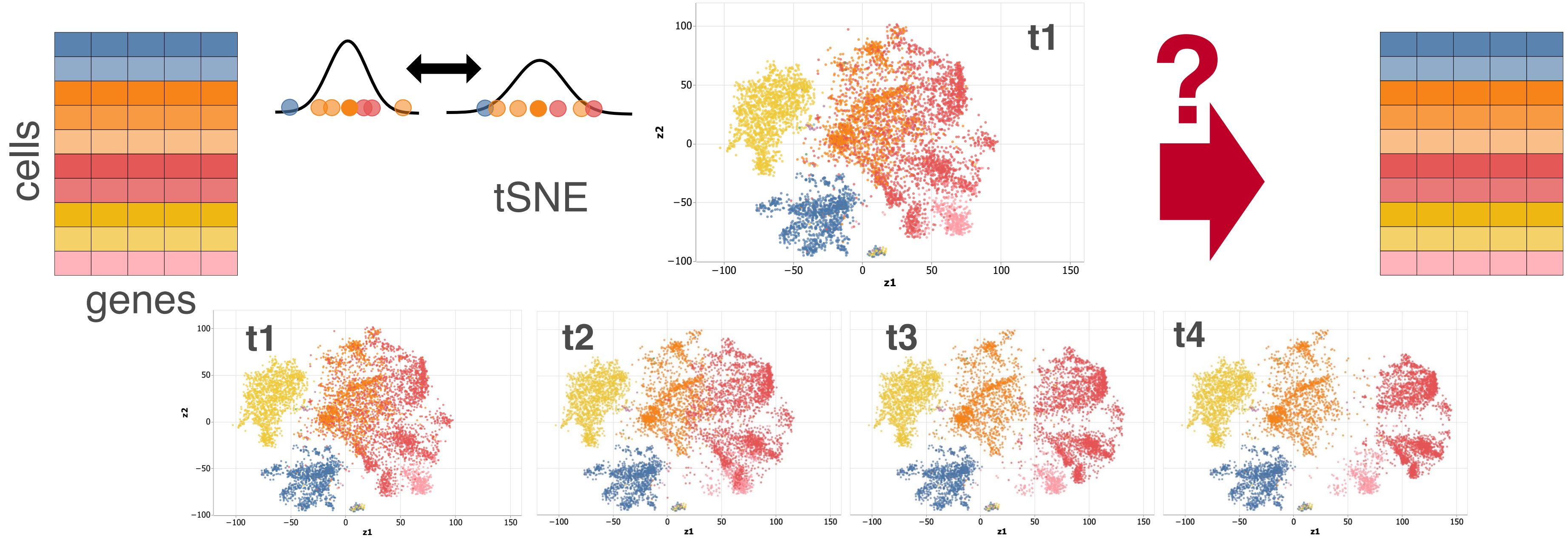
How to generate high-dimensional data from the low-dimensional time series?



1. Apply tSNE (or PCA/UMAP) to a snapshot scRNA-seq dataset

Inducing temporal patterns in tSNE-space

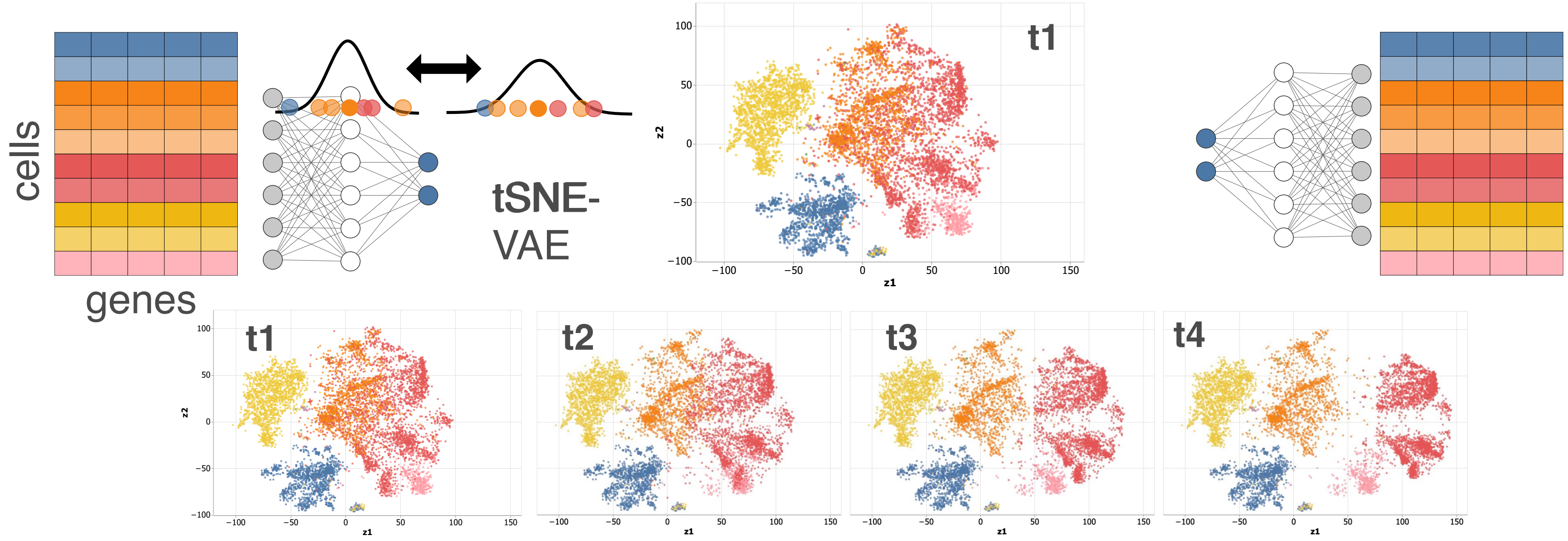
How to generate high-dimensional data from the low-dimensional time series?



1. Apply tSNE (or PCA/UMAP) to a snapshot scRNA-seq dataset

Inducing temporal patterns in tSNE-space

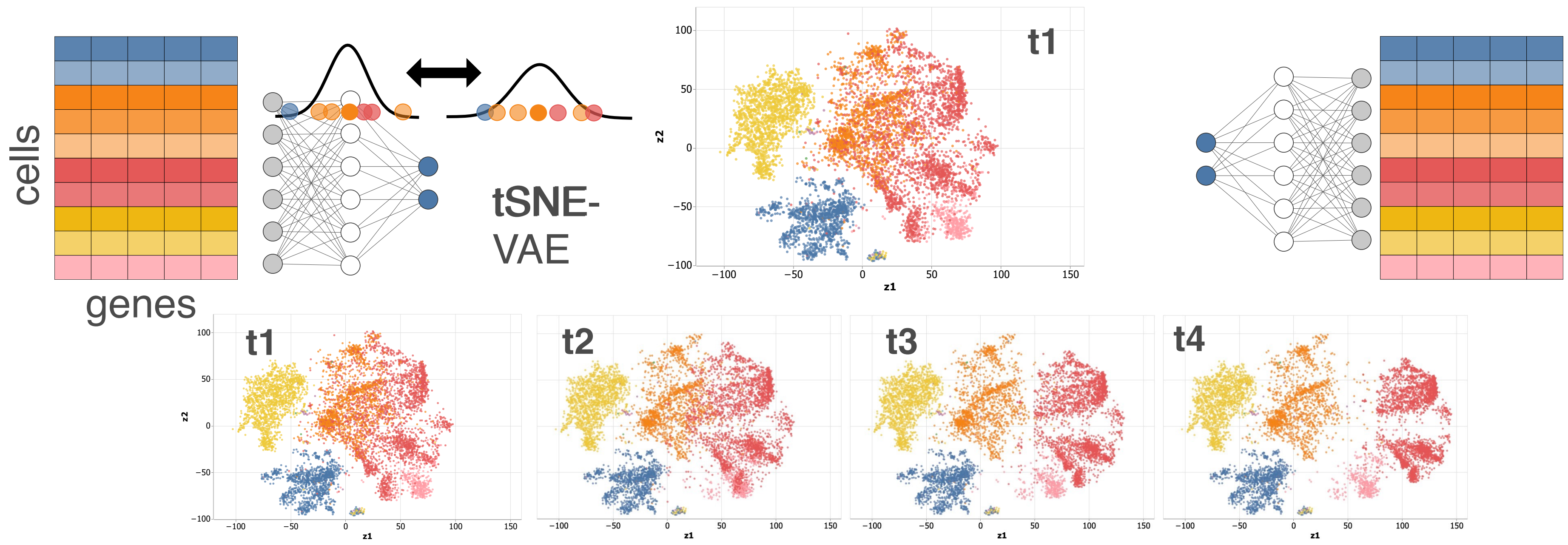
How to generate high-dimensional data from the low-dimensional time series?



1. Apply tSNE (or PCA/UMAP) to a snapshot scRNA-seq dataset
2. Train a VAE to learn the tSNE embedding in the encoder

Inducing temporal patterns in tSNE-space

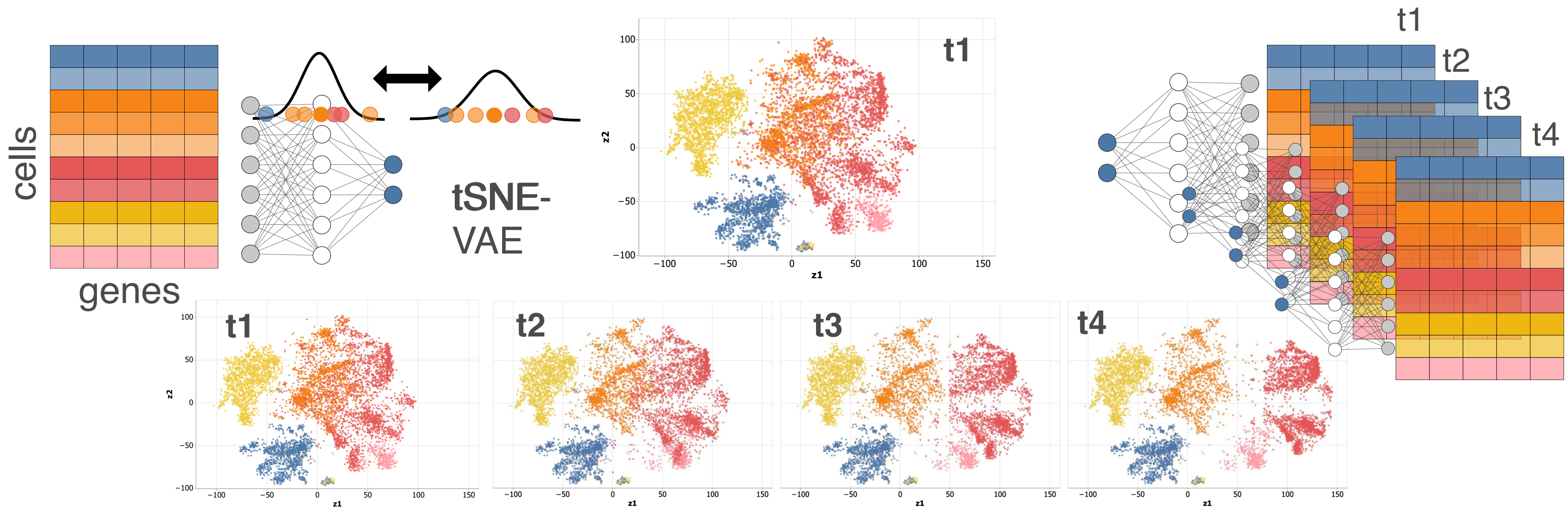
How to generate high-dimensional data from the low-dimensional time series?



1. Apply tSNE (or PCA/UMAP) to a snapshot scRNA-seq dataset
2. Train a VAE to learn the tSNE embedding in the encoder
3. In the low-dimensional space, induce a synthetic temporal pattern by vector field transformations

Inducing temporal patterns in tSNE-space

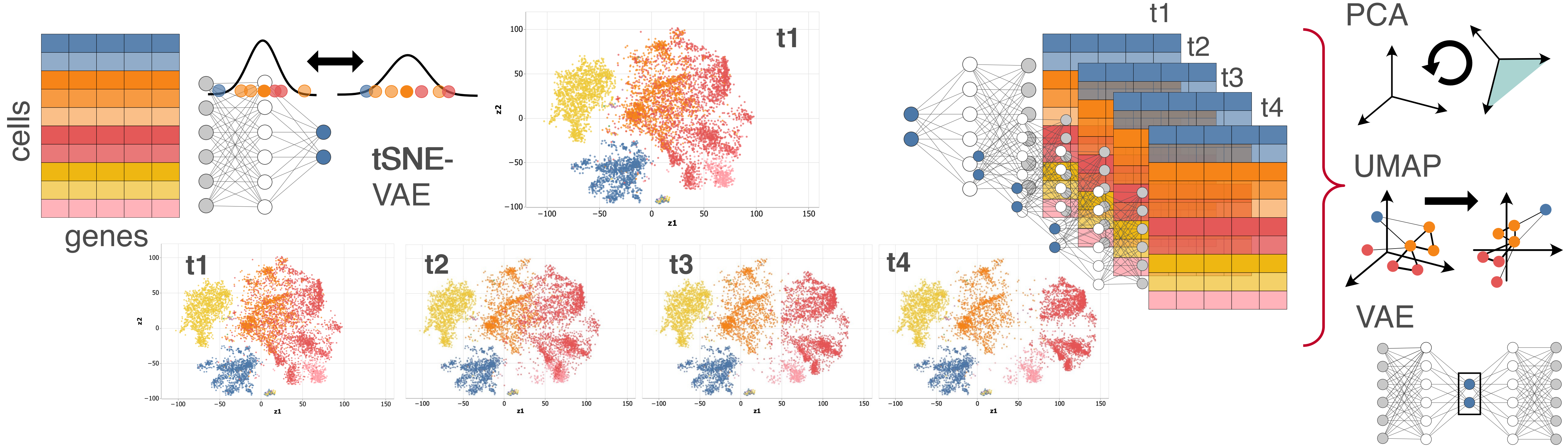
How to generate high-dimensional data from the low-dimensional time series?



1. Apply tSNE (or PCA/UMAP) to a snapshot scRNA-seq dataset
2. Train a VAE to learn the tSNE embedding in the encoder
3. In the low-dimensional space, induce a synthetic temporal pattern by vector field transformations
4. Use the decoder to generate corresponding high-dimensional data for each time point

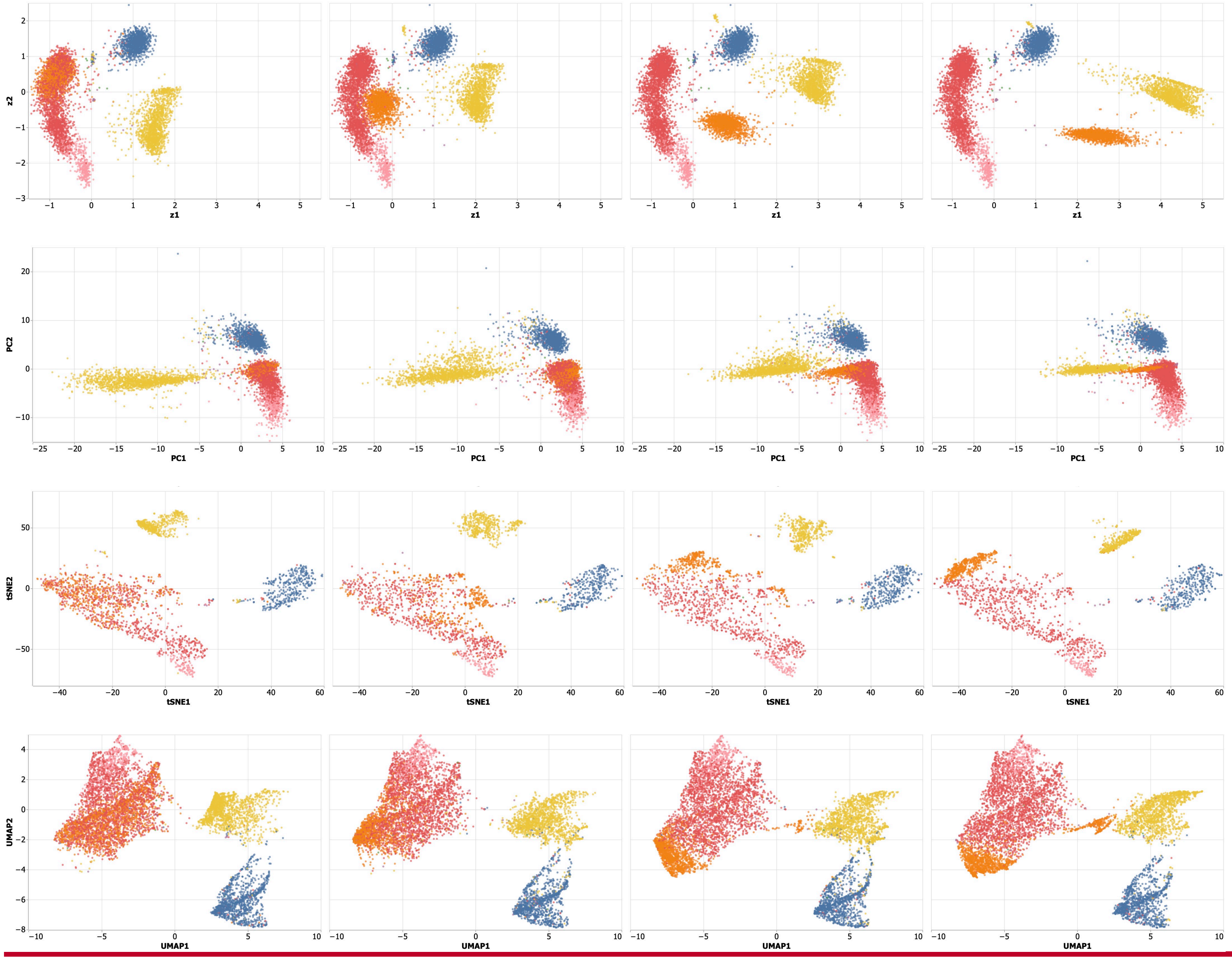
Inducing temporal patterns in tSNE-space

How to generate high-dimensional data from the low-dimensional time series?

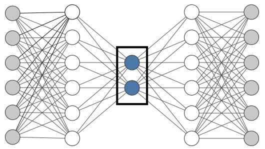


1. Apply tSNE (or PCA/UMAP) to a snapshot scRNA-seq dataset
2. Train a VAE to learn the tSNE embedding in the encoder
3. In the low-dimensional space, induce a synthetic temporal pattern by vector field transformations
4. Use the decoder to generate corresponding high-dimensional data for each time point
5. Apply PCA, UMAP and the VAE on the synthetic high-dimensional time-series data

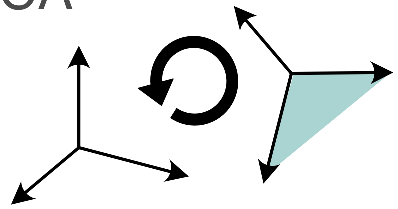
Results: Cluster spreading + rotation



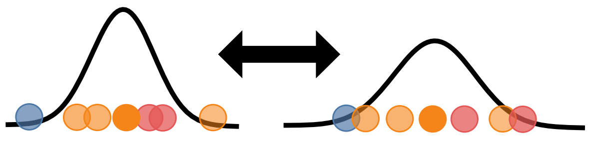
Orignal trans-
formation in the
VAE latent space



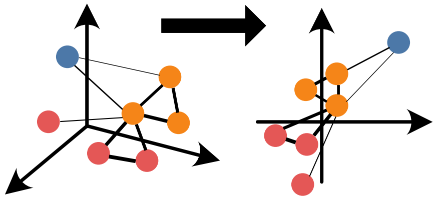
PCA



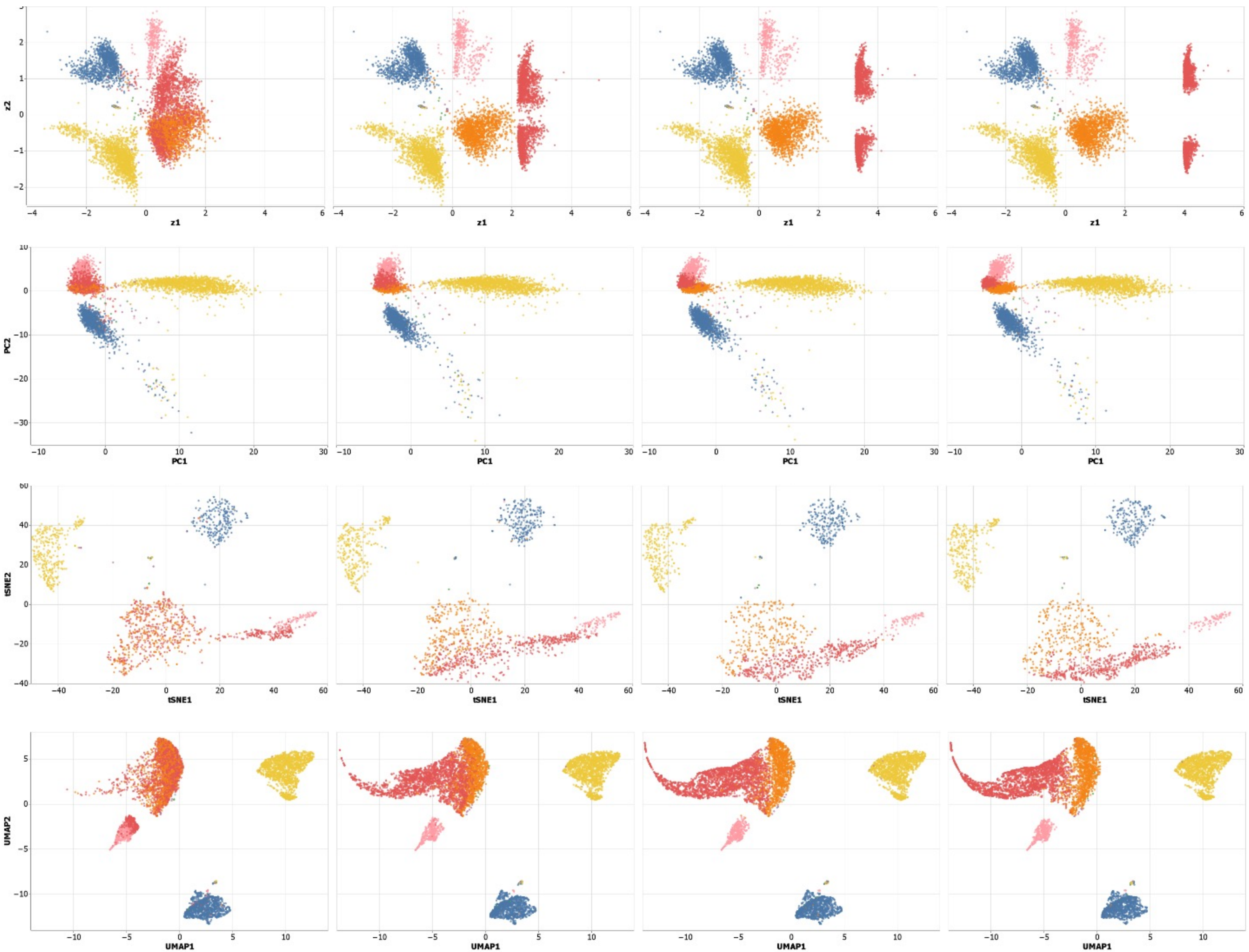
tSNE



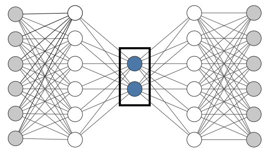
UMAP



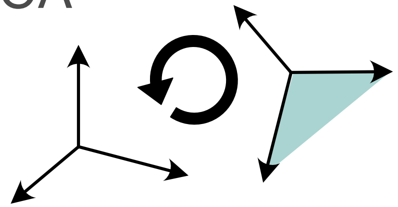
Results: Cluster division in VAE space



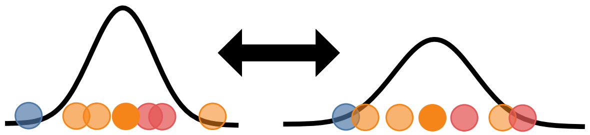
Orignal trans-
formation in the
VAE latent space



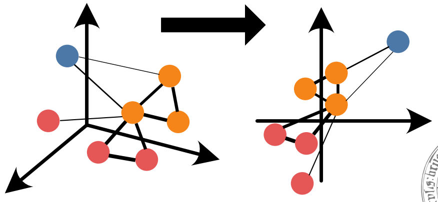
PCA



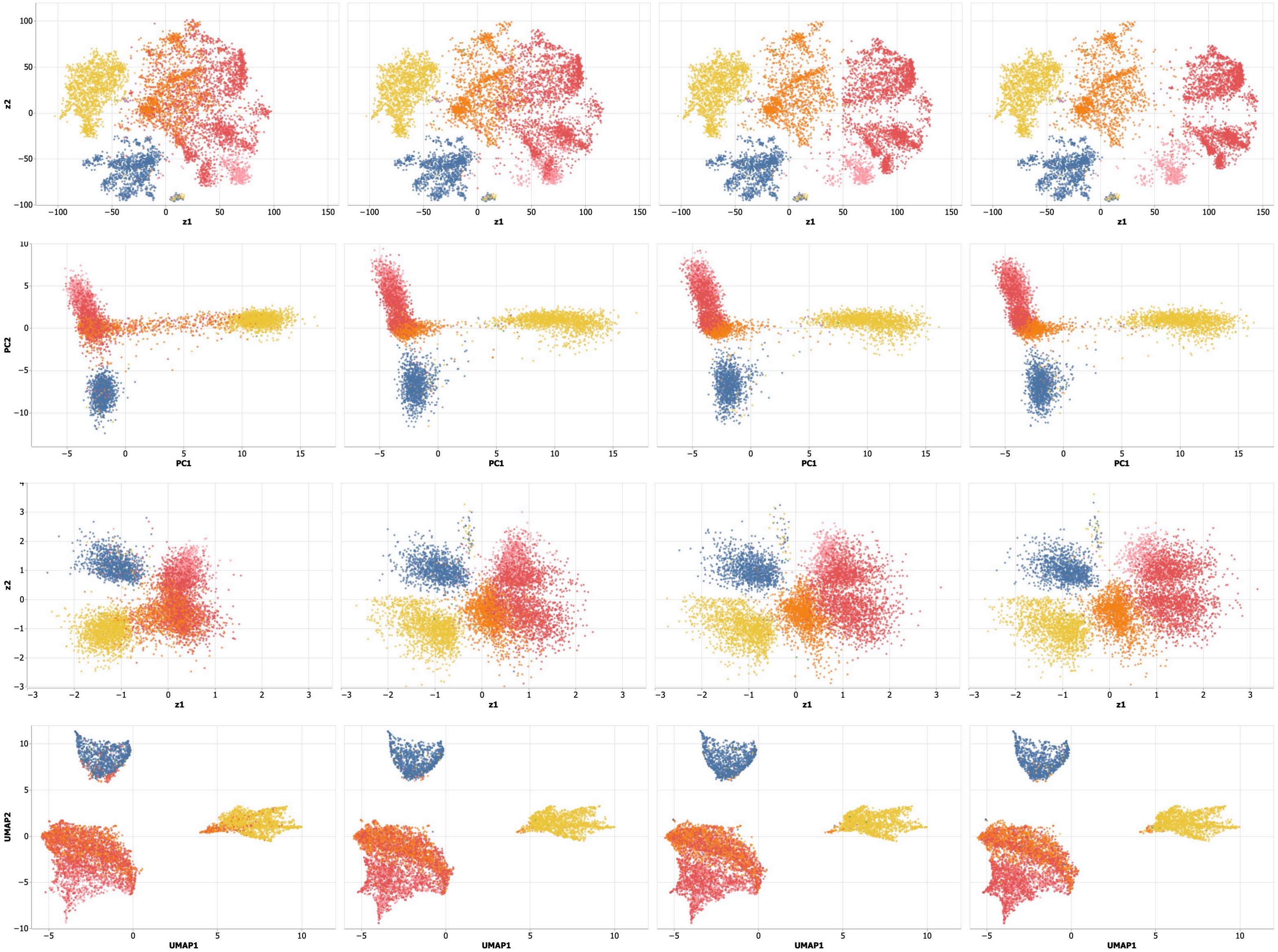
tSNE



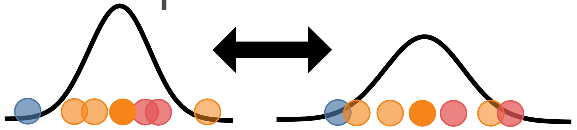
UMAP



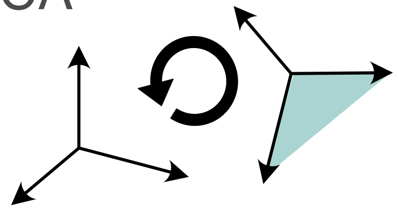
Results: Cluster division in tSNE space



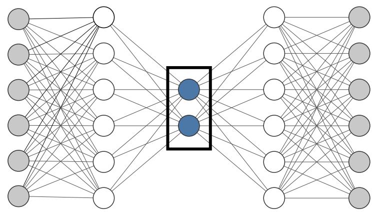
Orignal transformation in tSNE space



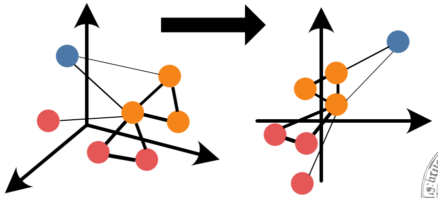
PCA



VAE



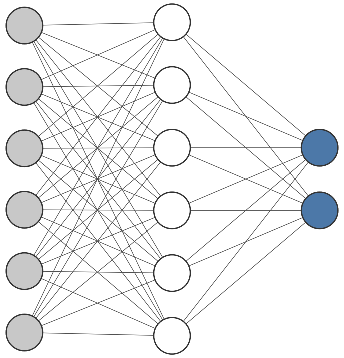
UMAP

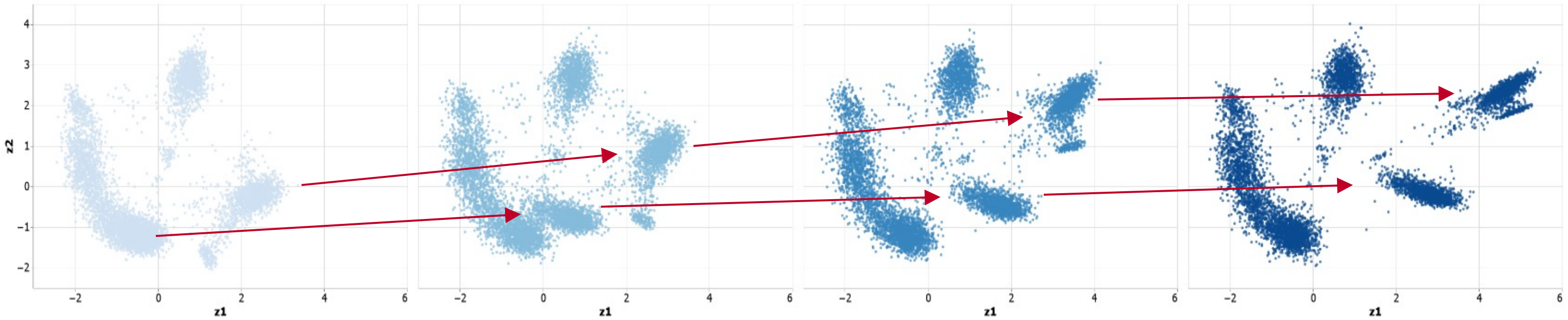


More than meets the eye…

- Popular dimension reduction techniques are not specifically designed to capture temporal patterns and do not necessarily reflect underlying dynamics
- Different dimension reduction methods have different perspectives on temporal devlopment patterns
- → Use these insights to design dimension reduction techniques that explicitly respect time structure
- → Have some explicit model for the temporal dynamics and let the model and the dimension reduction "talk to each other"

f_{encoder} ⟺ f_{dynamics}

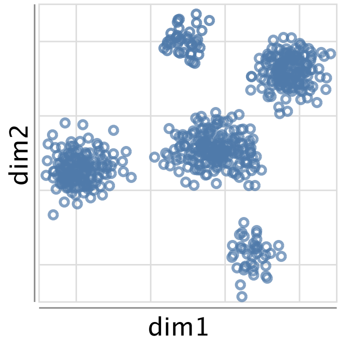




Why deep learning?

Neural networks can easily be combined with other explicit models

$x \in \mathbb{R}^p$ | tSNE algorithm | $z \in \mathbb{R}^2$

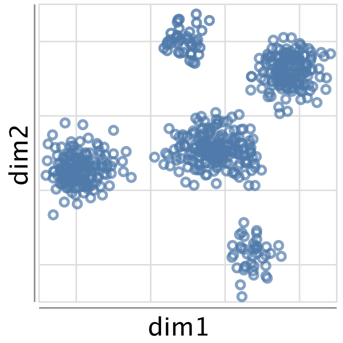


no explicit function f_{tSNE}

Why deep learning?

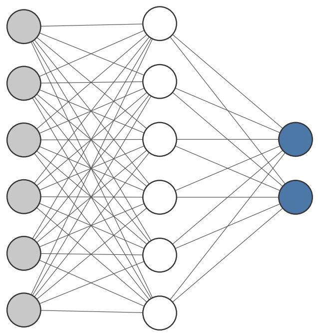
Neural networks can easily be combined with other explicit models

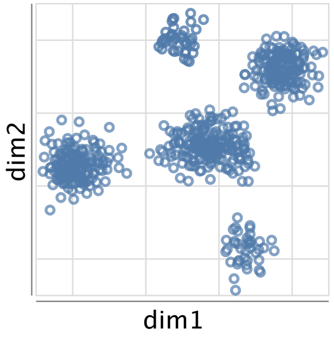
$x \in \mathbb{R}^p$ | tSNE algorithm | $z \in \mathbb{R}^2$

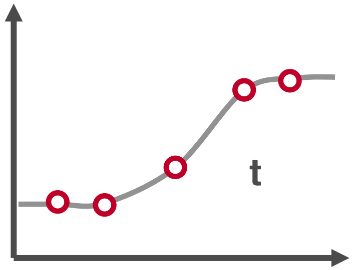


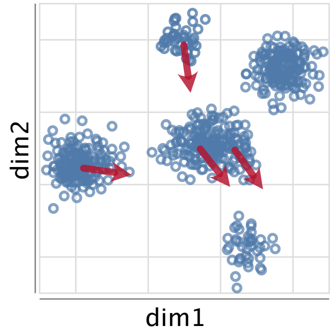
no explicit function f_{tSNE}

$x \in \mathbb{R}^p$ → VAE encoder f_{enc} → $z \in \mathbb{R}^2$ → dynamic model f_{dyn} → $z^{t_1}, z^{t_2}, \ldots \in \mathbb{R}^2$





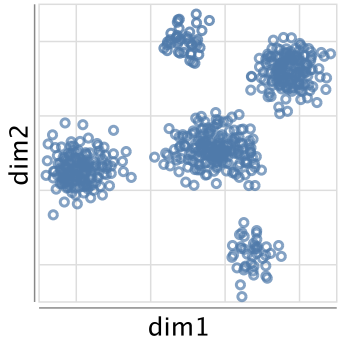




Why deep learning?

Neural networks can easily be combined with other explicit models

$x \in \mathbb{R}^p$ → tSNE algorithm → $z \in \mathbb{R}^2$

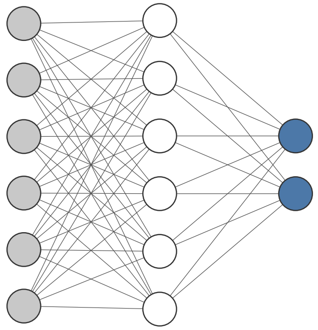


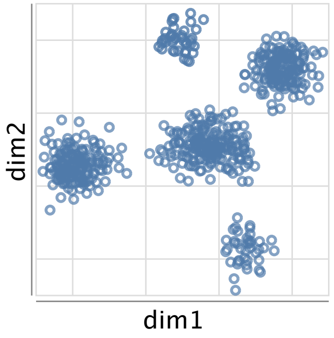
no explicit function f_{tSNE}

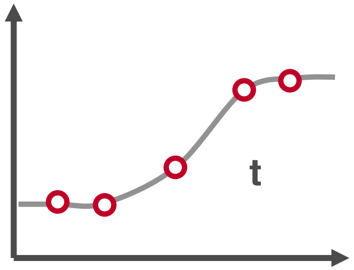
$x \in \mathbb{R}^p$ → VAE encoder f_{enc} → $z \in \mathbb{R}^2$ → dynamic model f_{dyn} → $z^{t_1}, z^{t_2}, \ldots \in \mathbb{R}^2$

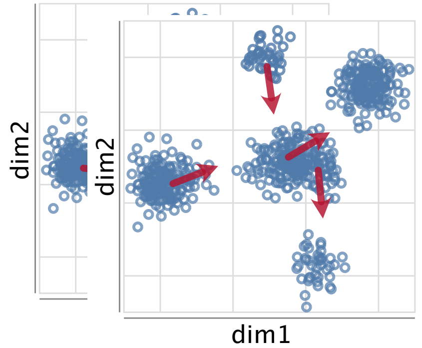
Define $\mathrm{loss}(x; \theta_{\mathrm{enc}}, \theta_{\mathrm{dyn}})$ and minimize w.r.t. $\theta_{\mathrm{enc}}, \theta_{\mathrm{dyn}}$

$\frac{\partial}{\partial \theta_{\mathrm{dyn}}} \mathrm{loss}$ ← $\frac{\partial}{\partial \theta} \mathrm{loss}(x; \theta_{\mathrm{enc}}, \theta_{\mathrm{dyn}})$





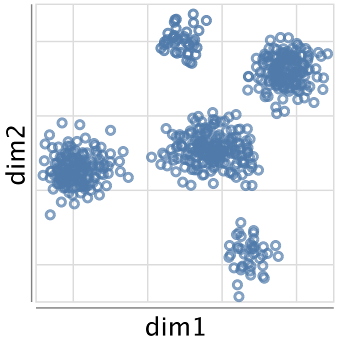




Why deep learning?

Neural networks can easily be combined with other explicit models

$x \in \mathbb{R}^p$ — tSNE algorithm — $z \in \mathbb{R}^2$

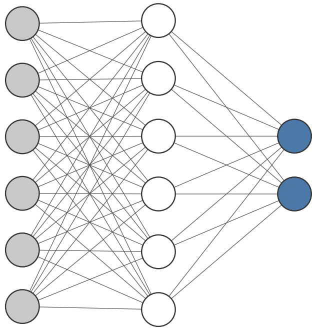


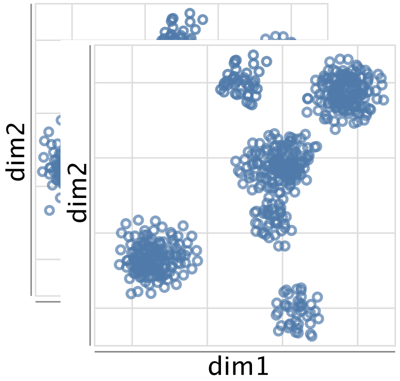
no explicit function f_{tSNE}

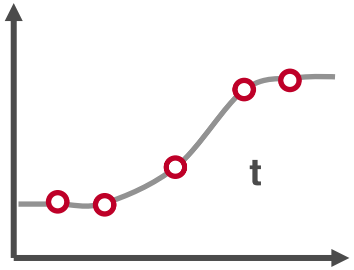
$x \in \mathbb{R}^p$ → VAE encoder f_{enc} → $z \in \mathbb{R}^2$ → dynamic model f_{dyn} → $z^{t_1}, z^{t_2}, \ldots \in \mathbb{R}^2$

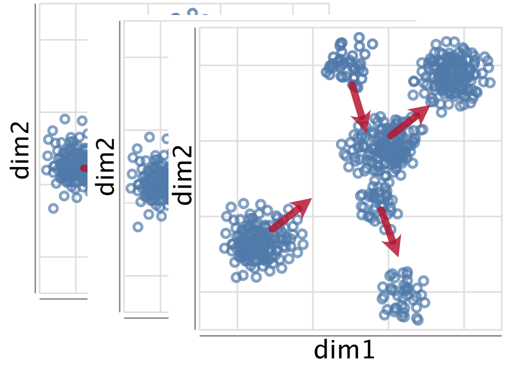
Define $\mathrm{loss}(x; \theta_{\mathrm{enc}}, \theta_{\mathrm{dyn}})$ and minimize w.r.t. $\theta_{\mathrm{enc}}, \theta_{\mathrm{dyn}}$

$\frac{\partial}{\partial \theta_{\mathrm{enc}}} \mathrm{loss}$ ← $\frac{\partial}{\partial \theta_{\mathrm{dyn}}} \mathrm{loss}$ ← $\frac{\partial}{\partial \theta} \mathrm{loss}(x; \theta_{\mathrm{enc}}, \theta_{\mathrm{dyn}})$









Differentiable programming to get the best of both worlds

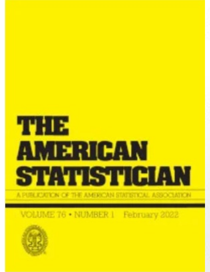
If you can program it, you can estimate the model parameters!

```julia
using Zygote

function loss(x, θ_enc, θ_dec, θ_dyn)
    z = VAE.enc(x, θ_enc)
    z' = dynamic_model(z, θ_dyn)
    x' = VAE.dec(z', θ_dec)
    lossval = ...
    return lossval
end

params = params(θ_enc, θ_dec, θ_dyn)
optimizer = ADAM(η)

for steps in 1:nsteps
    ∇loss = gradient(θ -> loss(x, θ...), params)
    update!(optimizer, params, ∇loss)
end
```

Zygote

Yann LeCun
January 5, 2018 ·

OK, Deep Learning has outlived its usefulness as a buzz-phrase.
Deep Learning est mort. Vive Differentiable Programming!



Statistical Practice

Using Differentiable Programming for Flexible Statistical Modeling

Maren Hackenberg, Marlon Grodd, Clemens Kreutz, Martina Fischer, Janina Esins, Linus Grabenhenrich, Christian Karagiannidis & Harald Binder ...show less

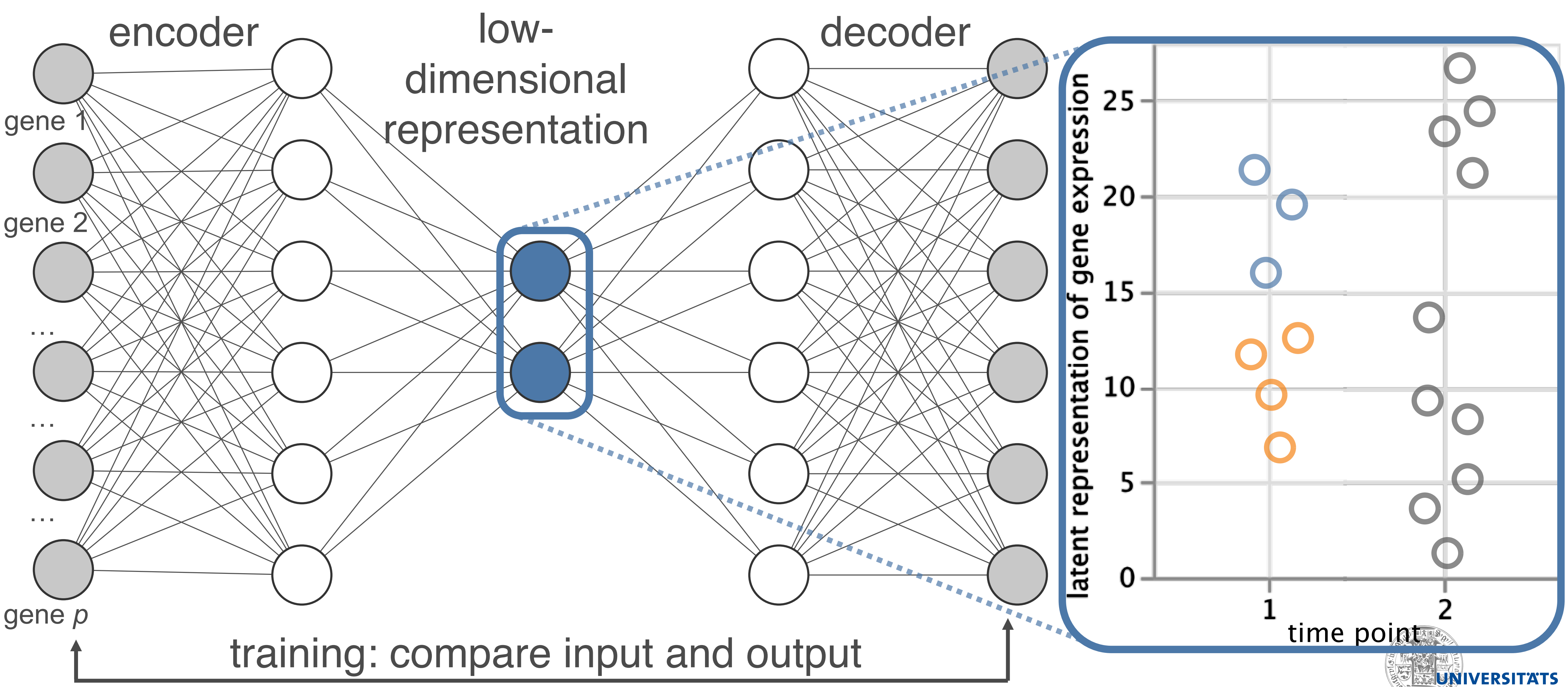
Received 07 Dec 2020, Accepted 17 Oct 2021, Accepted author version posted online: 09 Nov 2021, Published online: 21 Dec 2021

Download citation https://doi.org/10.1080/00031305.2021.2002189 Check for updates

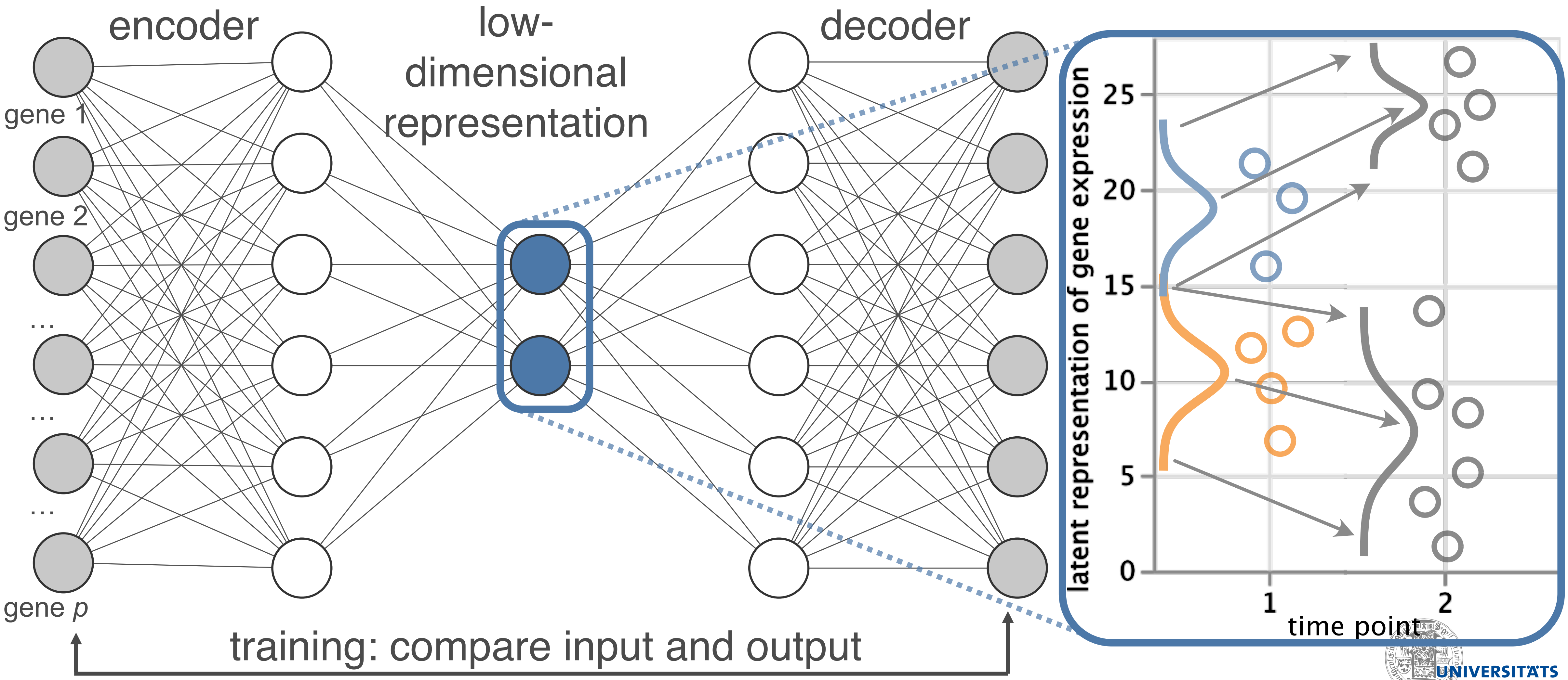
https://doi.org/10.1080/00031305.2021.2002189

UNIVERSITÄTS KLINIKUM FREIBURG

Combining VAEs with dynamic modeling for time-series single-cell RNA-sequencing

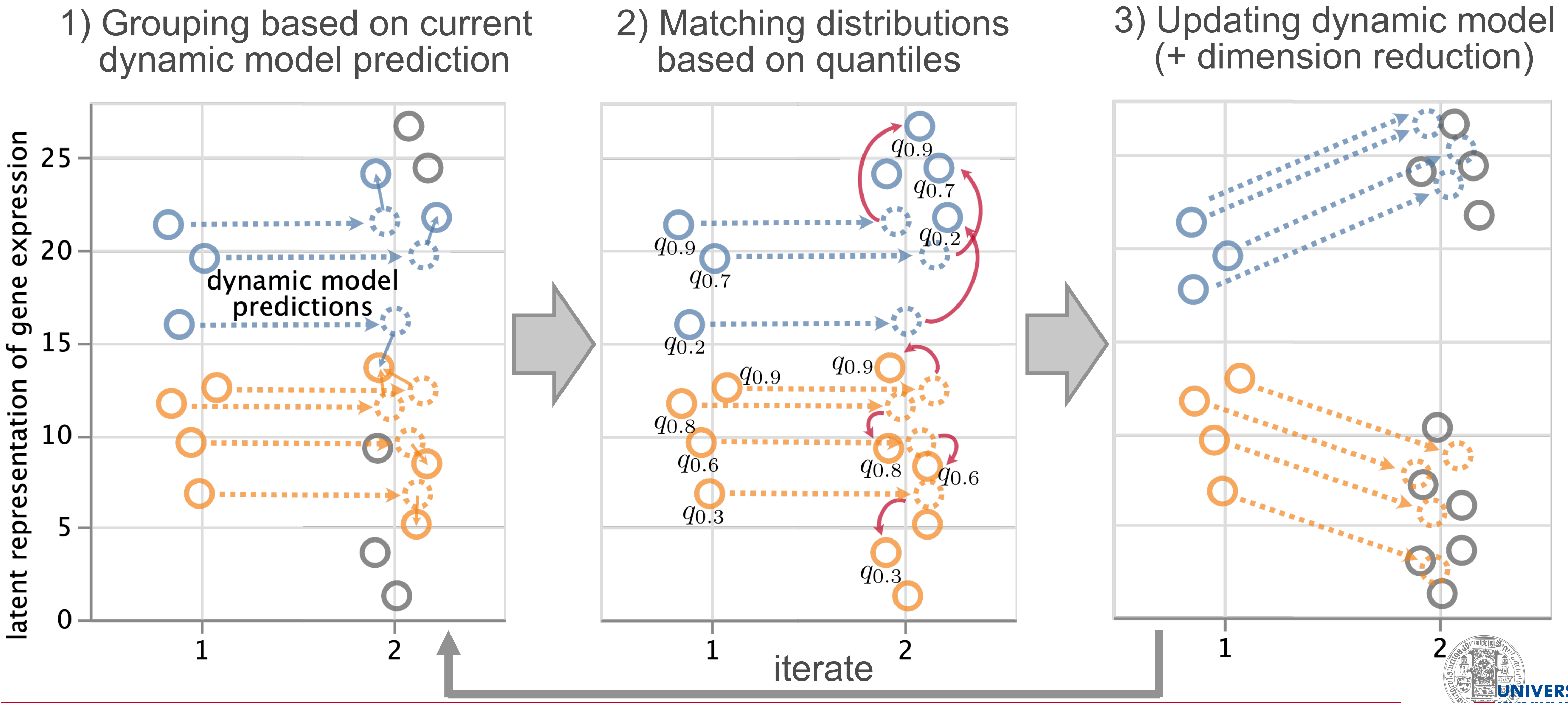


Combining VAEs with dynamic modeling for time-series single-cell RNA-sequencing



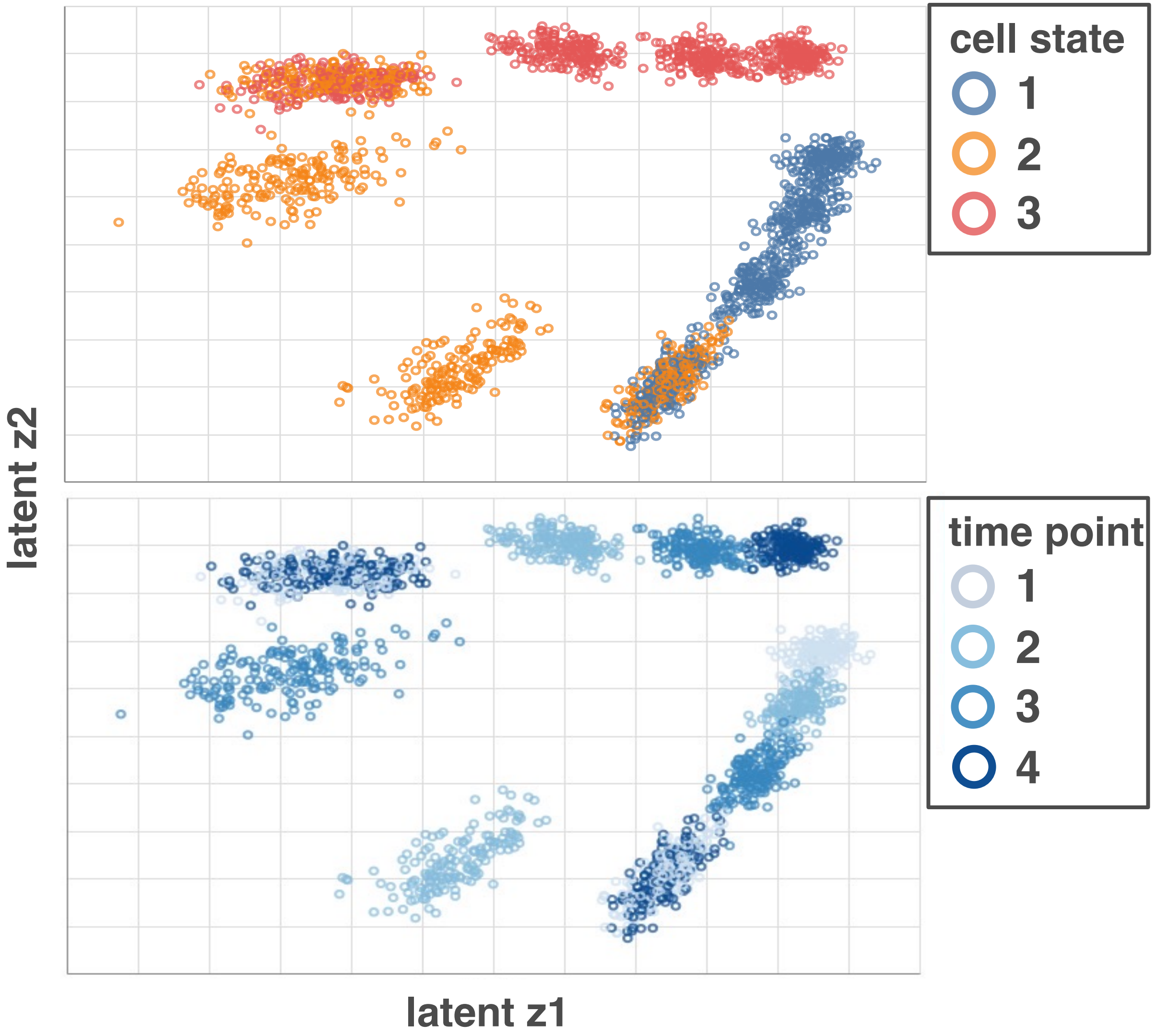
Modeling trajectories in a low-dimensional space

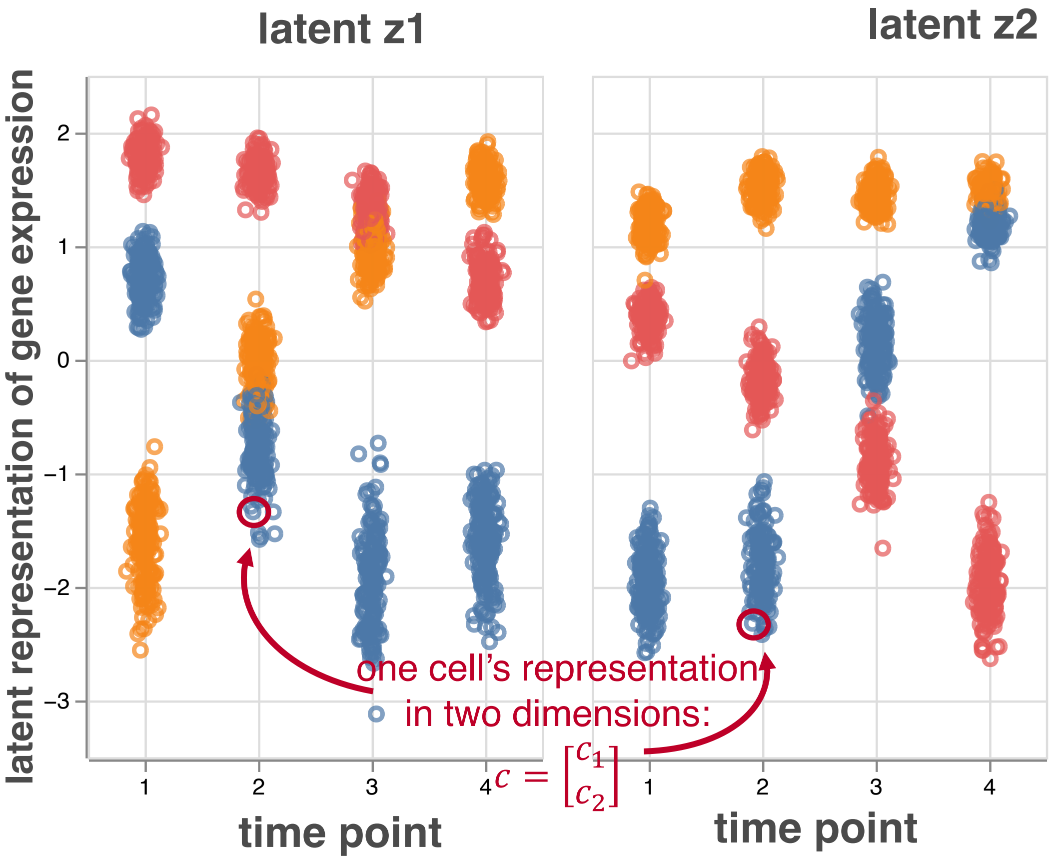
Iteratively optimize a dynamic model for describing trajectories in an EM framework



Simulating time-series scRNA-seq count data

Count data simulated with the `splatter` R package

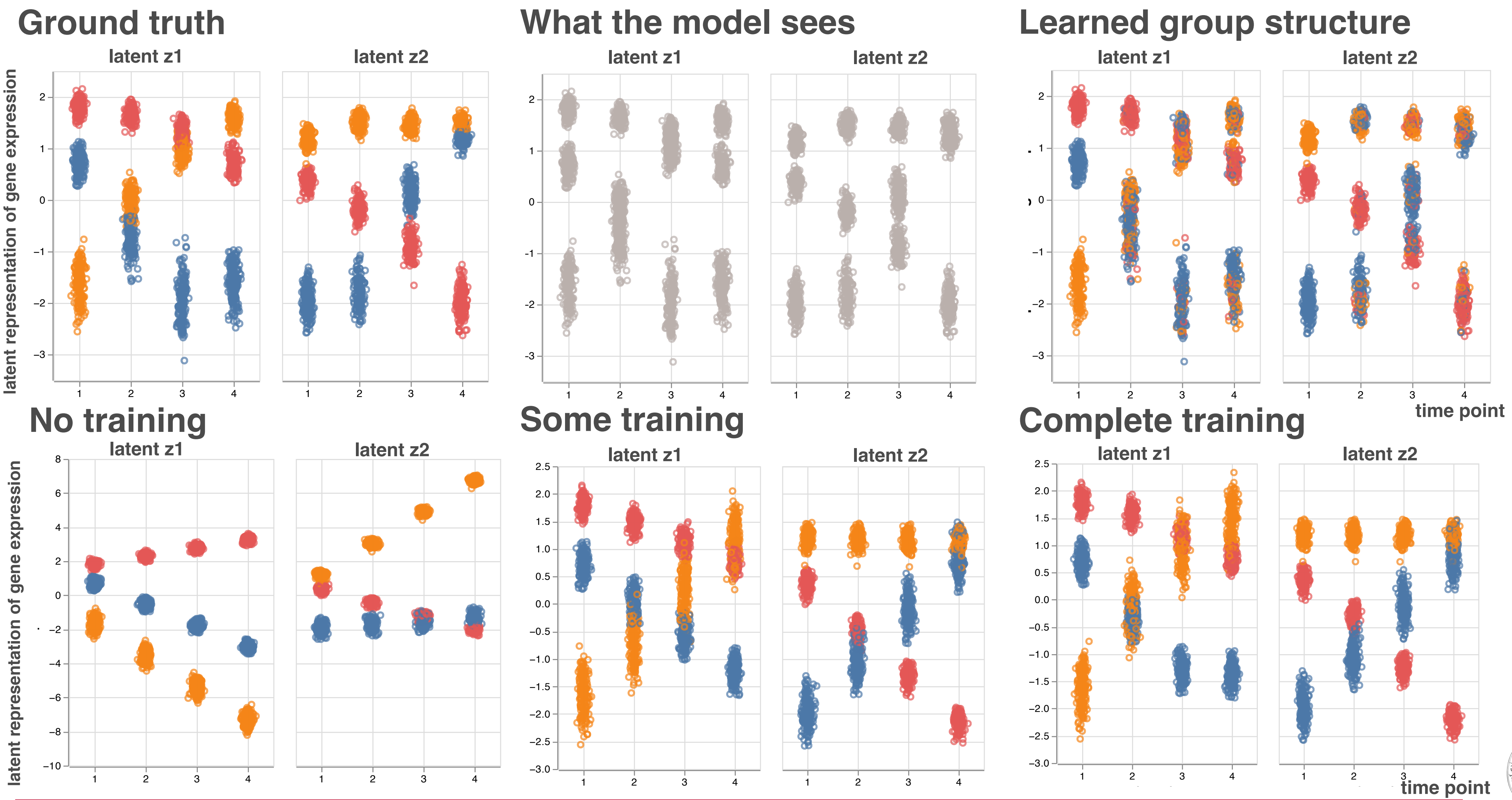




Each cell (circle) appears twice: dimension 1 of its low-dimensional representation in the left panel, dimension 2 in the right panel

Simulation study results

The dynamic model simultaneously infers underlying trajectories and cell states



VAE represen-tation of original data

Model predictions

Summary

- **time-series single-cell RNA-seq is promising…**
  - study highly-resolved subgroups of cells
  - understand group-specific developmental trajectories
- **… but also implies specific challenges:**
  - find a suitable dimension reduction that reflects temporal patterns
  - link groups of cells across time despite no one-to-one correspondence
- using **deep learning**-based approaches and combining them with dynamic **modeling** can help to address these challenges

Thanks to…

Harald Binder

Laia Canal Guitart

Martin Treppner

Kiana Farhadyar

Sara Al-Rawi

Moritz Hess

Göran Köber

Camila Fullio

Tanja Vogel

And thanks to you for listening!

Questions?

Please get in touch! maren@imbi.uni-freiburg.de