gmds Deutsche Gesellschaft für Medizinische Informatik, Biometrie und Epidemiologie e.V.

Summer Academy

23.-26.9.2024, Schloss Fürstenried, München

UNIVERSITÄTS KLINIKUM FREIBURG

small data MORE FROM LESS CRC 1597

Workshop 3 - Part 1
Use of AI and ML in Statistics, with a focus on application, interpretation and small data challenges

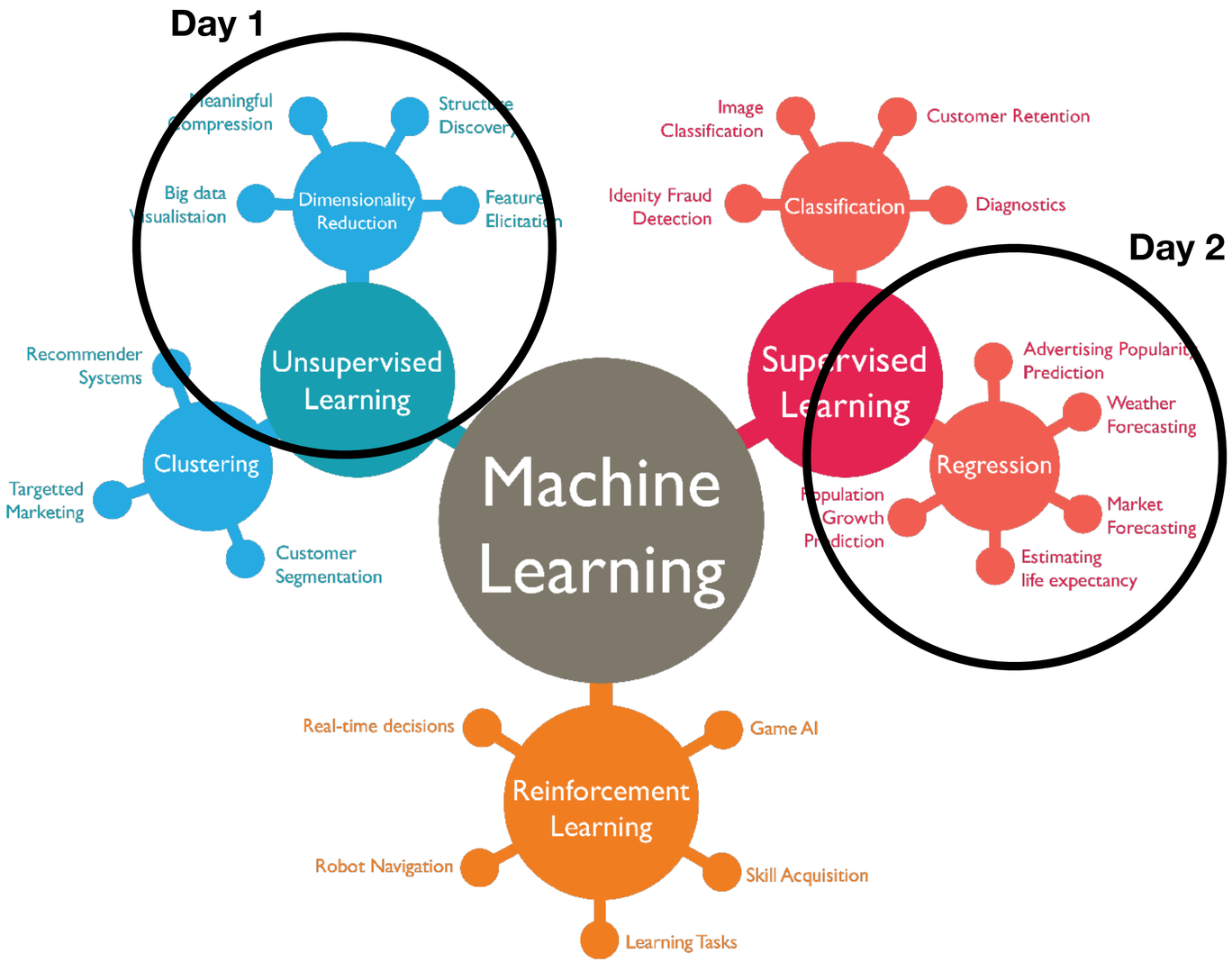
Institute of Medical Biometry and Statistics (IMBI)

Moritz Hess

⚠ **This course was created with the help of AI**

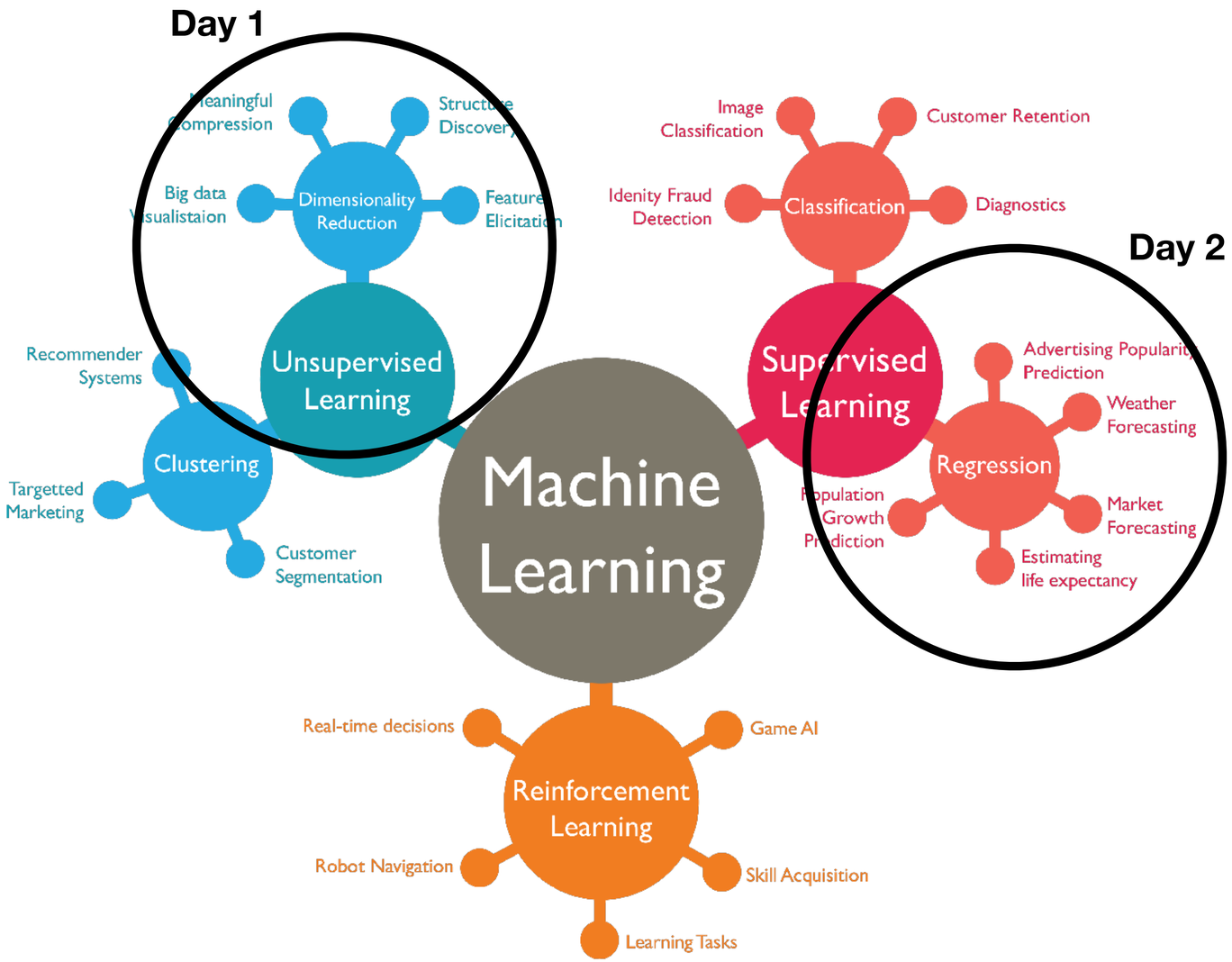
Two 1/4 day journey through machine learning / AI

Two 1/4 day journey through machine learning / AI

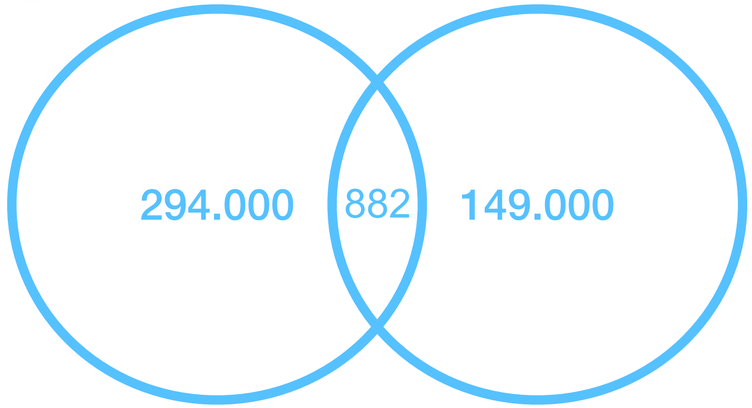


small data
MORE FROM LESS

Two 1/4 day journey through machine learning / AI



R (CRAN package) **PyTorch**



Google Scholar search 20.9.2024

small data
MORE FROM LESS

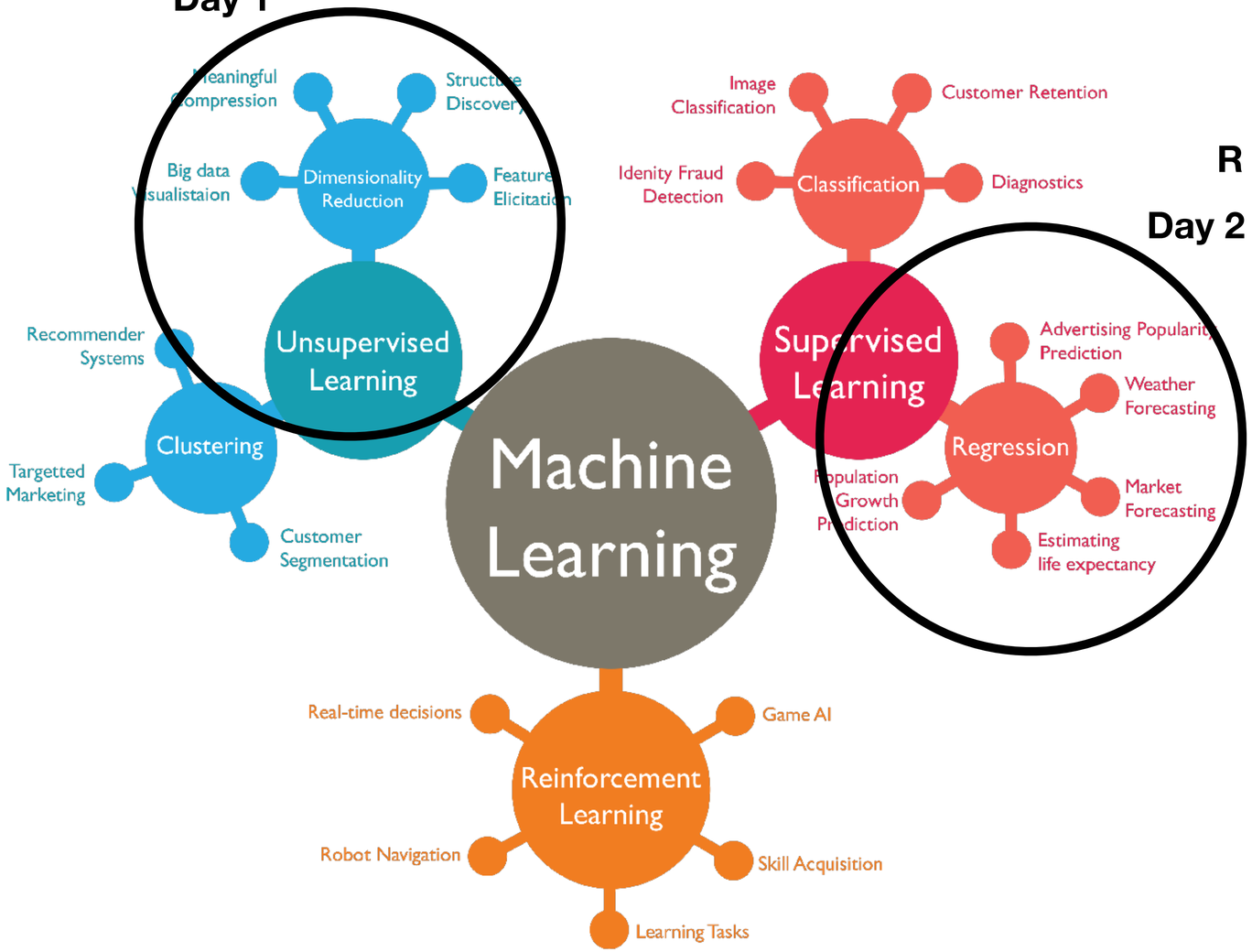
Two 1/4 day journey through machine learning / AI

Pytorch

Day 1

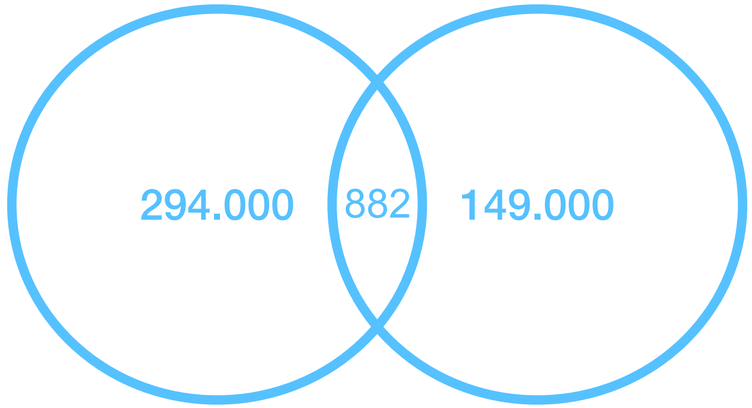
R

Day 2



R (CRAN package)

PyTorch

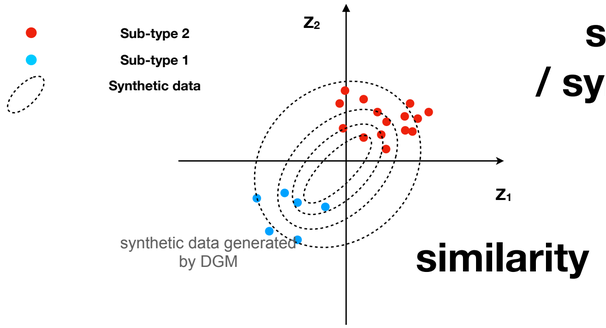


Google Scholar search 20.9.2024

small data
MORE FROM LESS

Me / IMBI / Small Data Initiative

Me / IMBI / Small Data Initiative



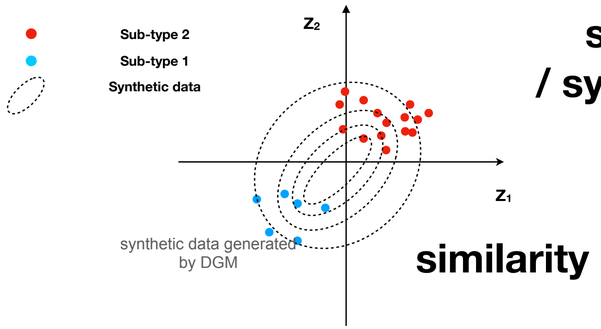
simulation / synthetic data

interpretation

similarity

small data
MORE FROM LESS

Me / IMBI / Small Data Initiative



simulation / synthetic data

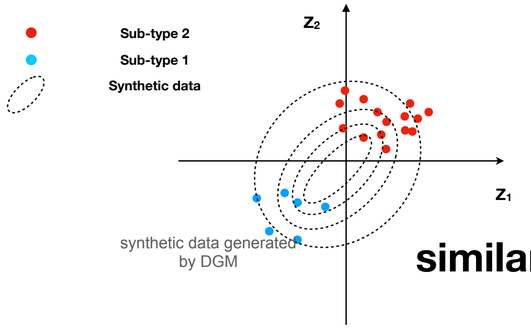
interpretation

similarity

https://www.uniklinik-freiburg.de/imbi.html

small data
MORE FROM LESS

Me / IMBI / Small Data Initiative

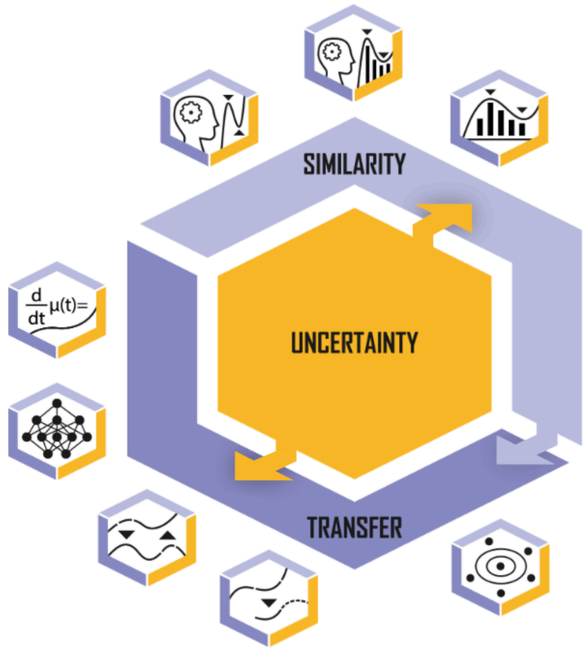


simulation / synthetic data

interpretation

similarity

https://www.uniklinik-freiburg.de/imbi.html



https://www.smalldata-initiative.de

small data
MORE FROM LESS

You

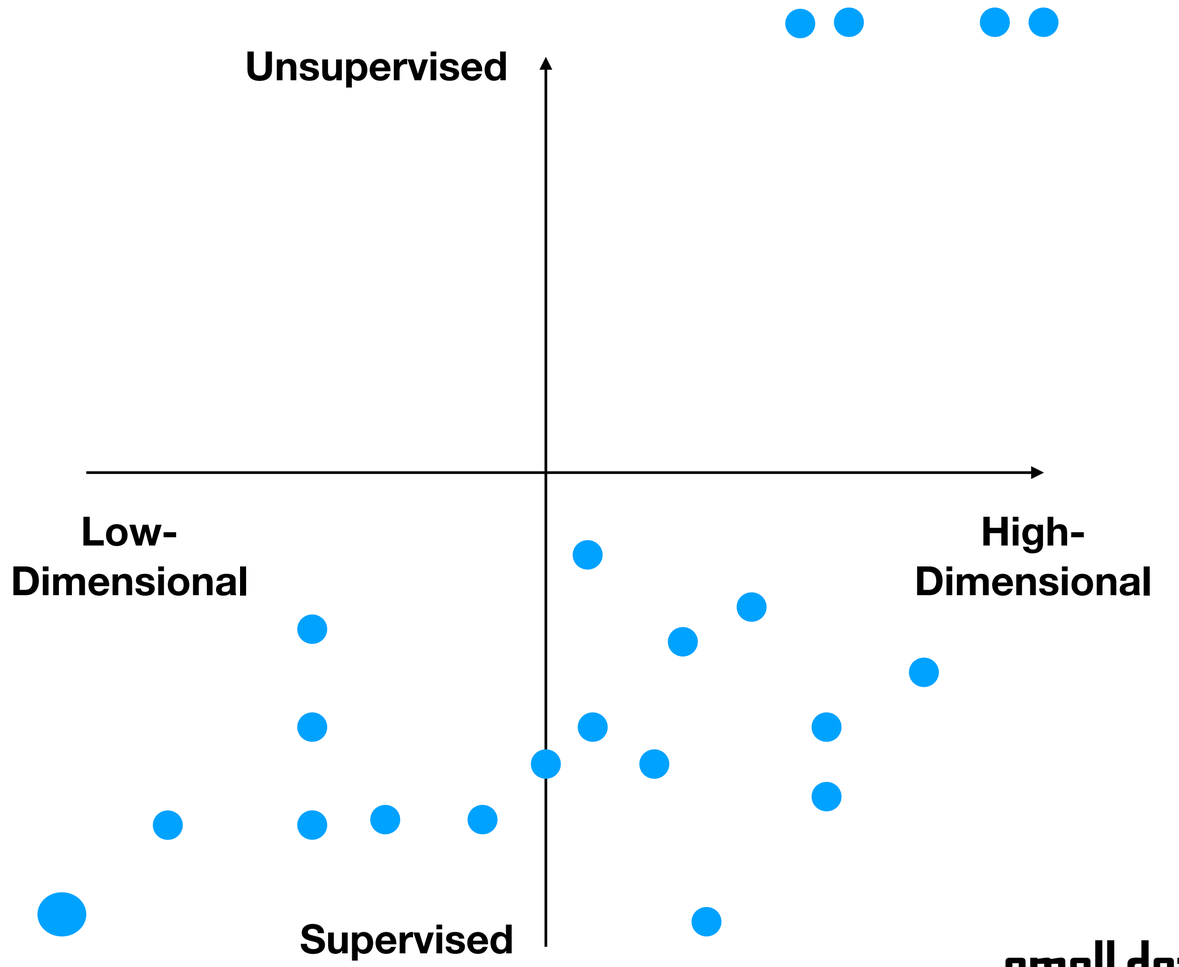
Expectations

What are your expectations / interests?

Knowledge
- **Statistics / probability theory**
- **Computer science**

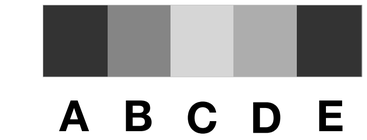
Abilities
- **Using the right approach for your research**
- **Programming**
- **Interpreting the output of an approach**

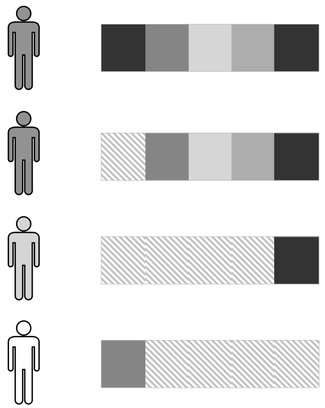
Typical data and approach



small data
MORE FROM LESS

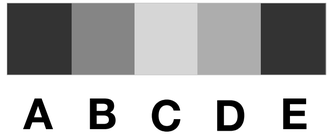
Introduction: AI ↔ Neural networks

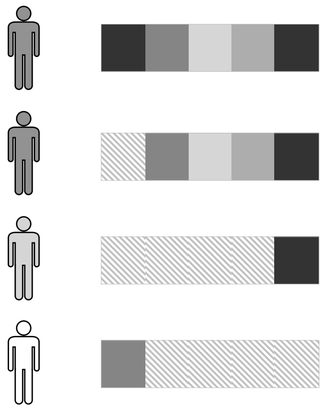




Observations

Introduction: AI ↔ Neural networks





Observations

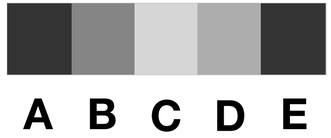
$$y = \beta_0 + \beta_1 A + \ldots + \beta_5 E + \epsilon$$

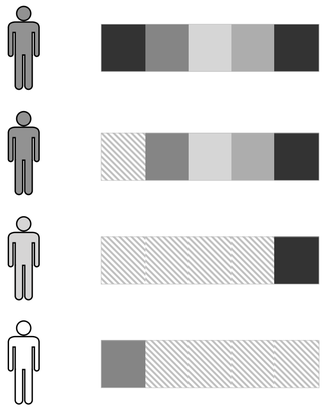
$$y = \beta_0 + \beta_1 A + \ldots + \beta_5 E + \beta_6 AB + \epsilon$$

$$y = \beta_0 + f_1(A) + \ldots + f_5(E) + \epsilon$$

Traditional

Introduction: AI ↔ Neural networks





Observations

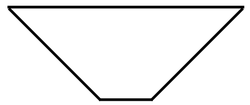
$$y = \beta_0 + \beta_1 A + \ldots + \beta_5 E + \epsilon$$

$$y = \beta_0 + \beta_1 A + \ldots + \beta_5 E + \beta_6 AB + \epsilon$$

$$y = \beta_0 + f_1(A) + \ldots + f_5(E) + \epsilon$$

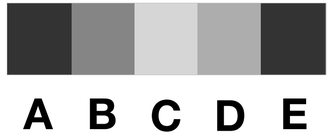
Traditional

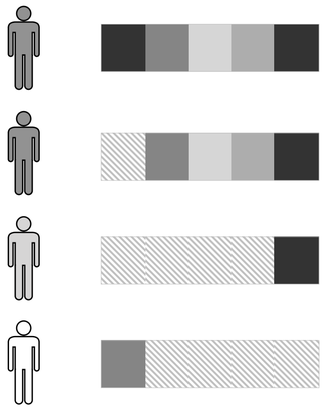
$$y = f_\theta(A, B, C, D, E) + \epsilon$$



“AI”

Introduction: AI ↔ Neural networks





Observations

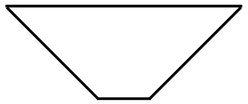
$$y = \beta_0 + \beta_1 A + \ldots + \beta_5 E + \epsilon$$

$$y = \beta_0 + \beta_1 A + \ldots + \beta_5 E + \beta_6 AB + \epsilon$$

$$y = \beta_0 + f_1(A) + \ldots + f_5(E) + \epsilon$$

Traditional

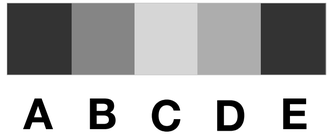
$$y = f_\theta(A, B, C, D, E) + \epsilon$$

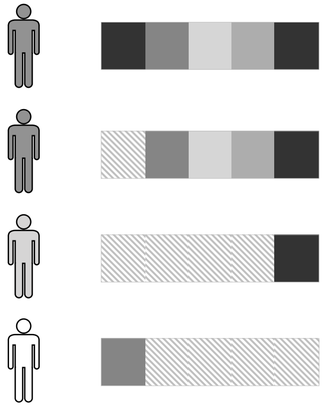


“AI”

Universal Function Approximator

Introduction: AI ↔ Neural networks





Observations

$$y = \beta_0 + \beta_1 A + \ldots + \beta_5 E + \epsilon$$

$$y = \beta_0 + \beta_1 A + \ldots + \beta_5 E + \beta_6 AB + \epsilon$$

$$y = \beta_0 + f_1(A) + \ldots + f_5(E) + \epsilon$$

Traditional

http://neuralnetworksanddeeplearning.com/chap4.html

$$y = f_\theta(A, B, C, D, E) + \epsilon$$

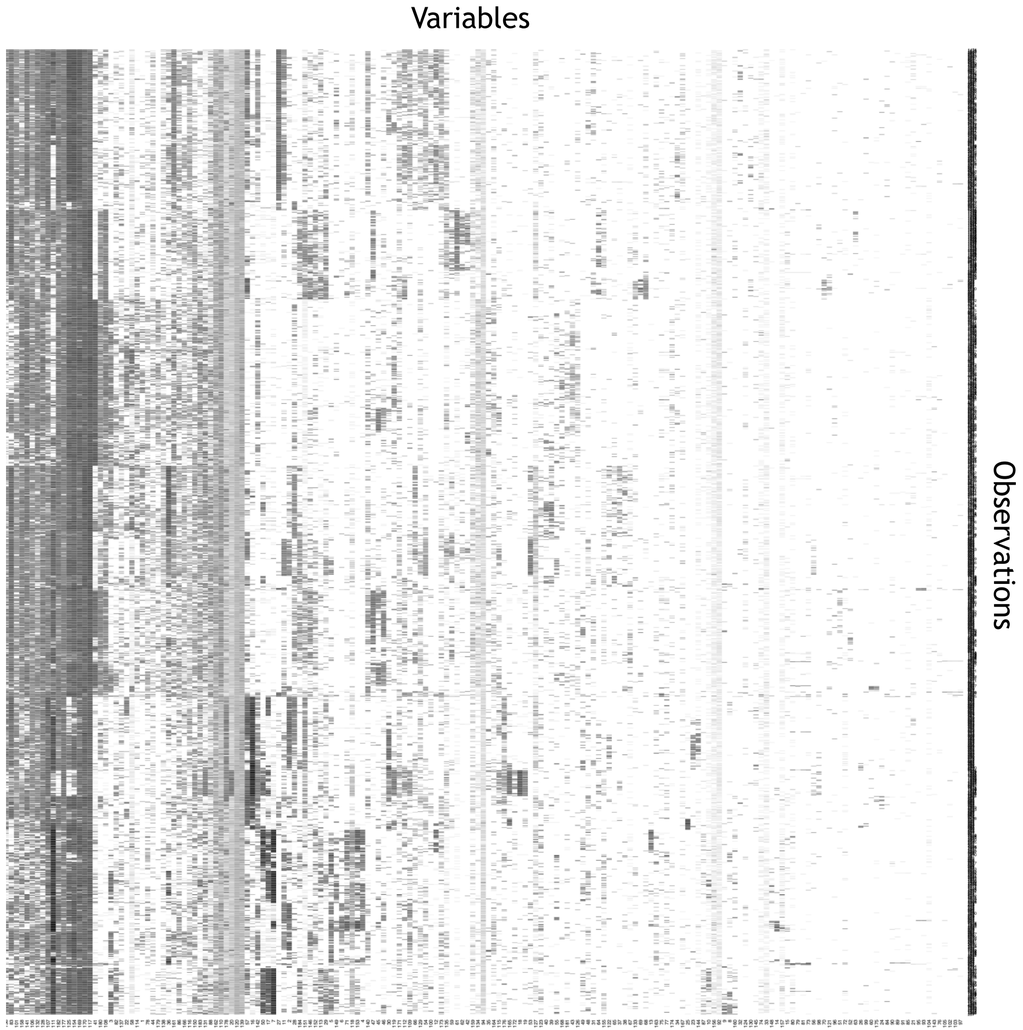
Universal Function Approximator

“AI”

small data
MORE FROM LESS

Part 1: Neural Networks
Lecture 45 min , practicals 30 min

Course of dimensionality - example



Gene expression investigated in single cells

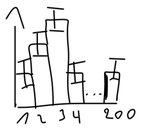
RESOURCE

nature neuroscience

Adult mouse cortical cell taxonomy revealed by single cell transcriptomics

Bosiljka Tasic[1,2], Vilas Menon[1,2], Thuc Nghi Nguyen[1], Tae Kyung Kim[1], Tim Jarsky[1], Zizhen Yao[1], Boaz Levi[1], Lucas T Gray[1], Staci A Sorensen[1], Tim Dolbeare[1], Darren Bertagnolli[1], Jeff Goldy[1], Nadiya Shapovalova[1], Sheana Parry[1], Changkyu Lee[1], Kimberly Smith[1], Amy Bernard[1], Linda Madisen[1], Susan M Sunkin[1], Michael Hawrylycz[1], Christof Koch[1] & Hongkui Zeng[1]

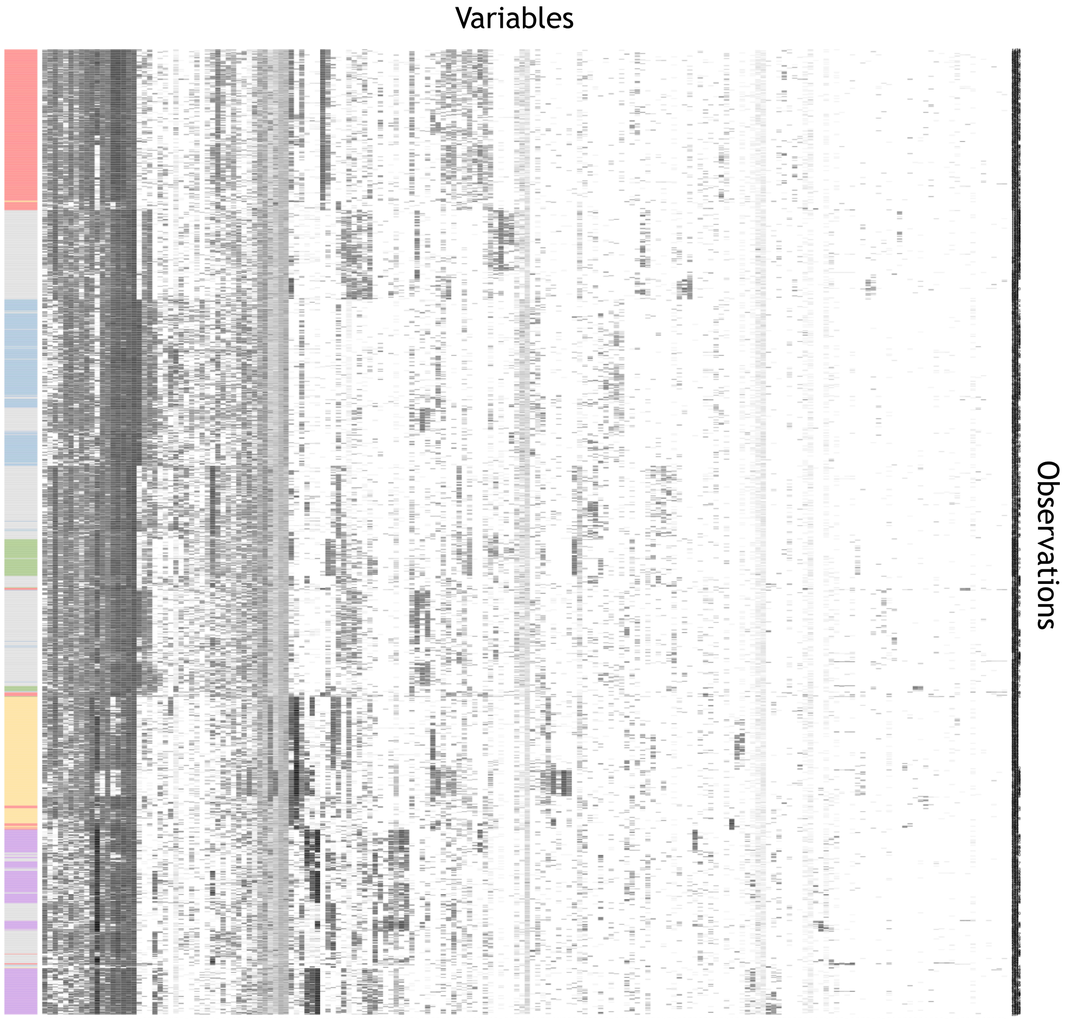
1525 observations



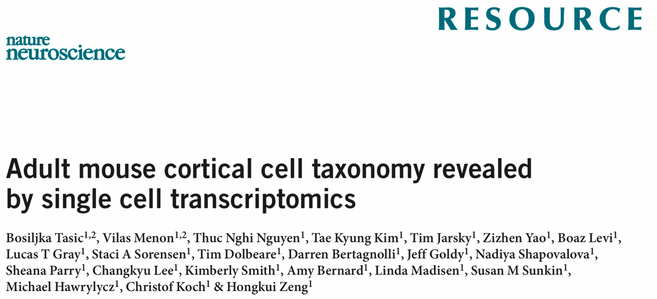
180 variables

small data
MORE FROM LESS

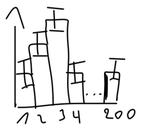
Course of dimensionality - example



Gene expression investigated in single cells



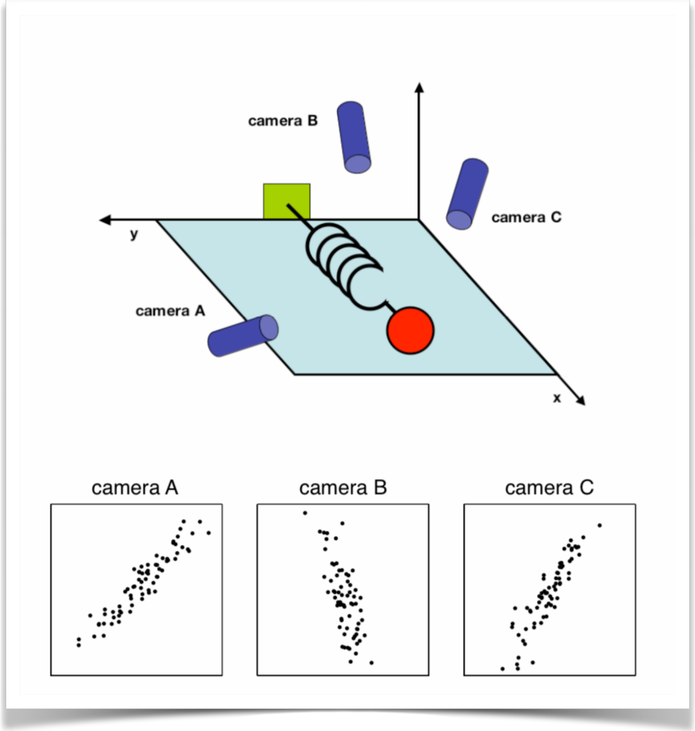
1525 observations



180 variables

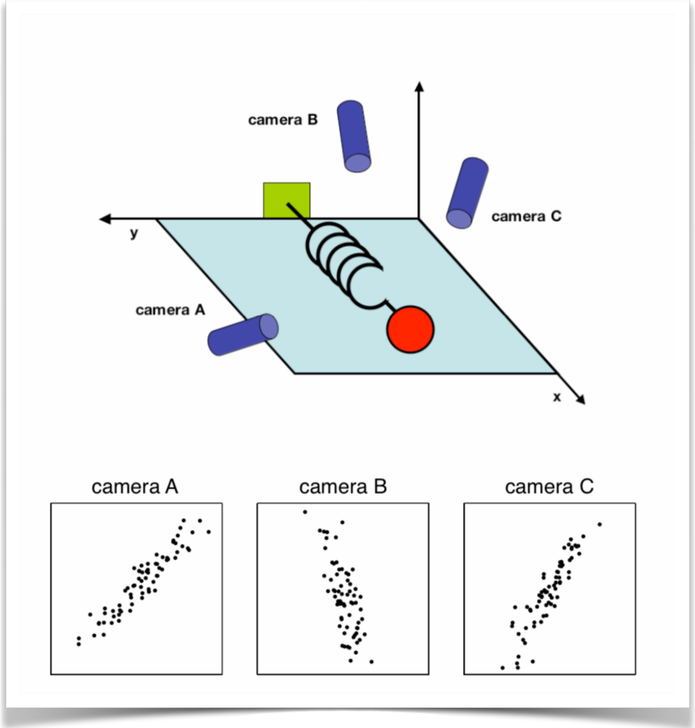
small data
MORE FROM LESS

Principal component analysis (PCA)

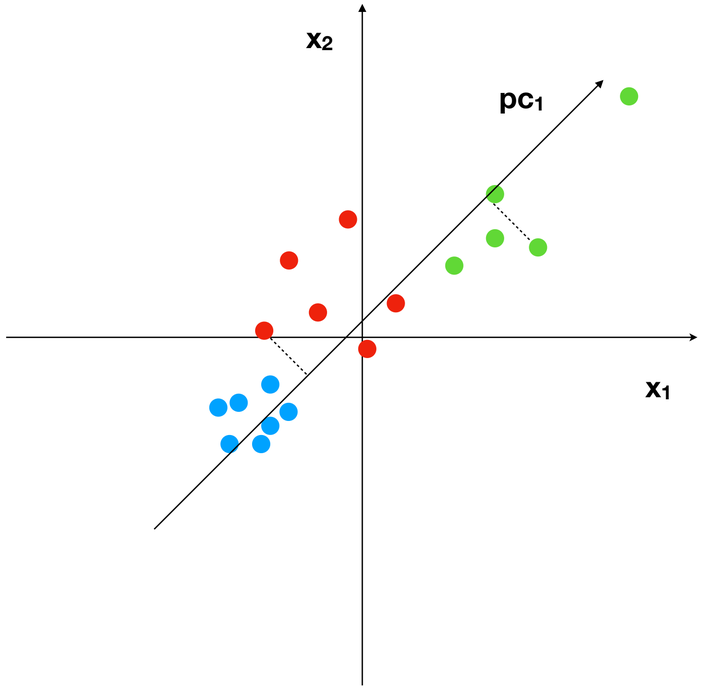


Shlens et al. 2014

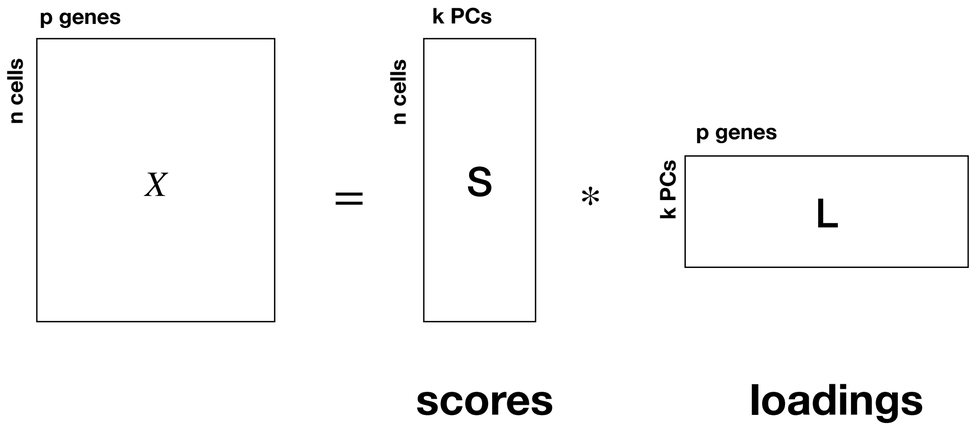
Principal component analysis (PCA)

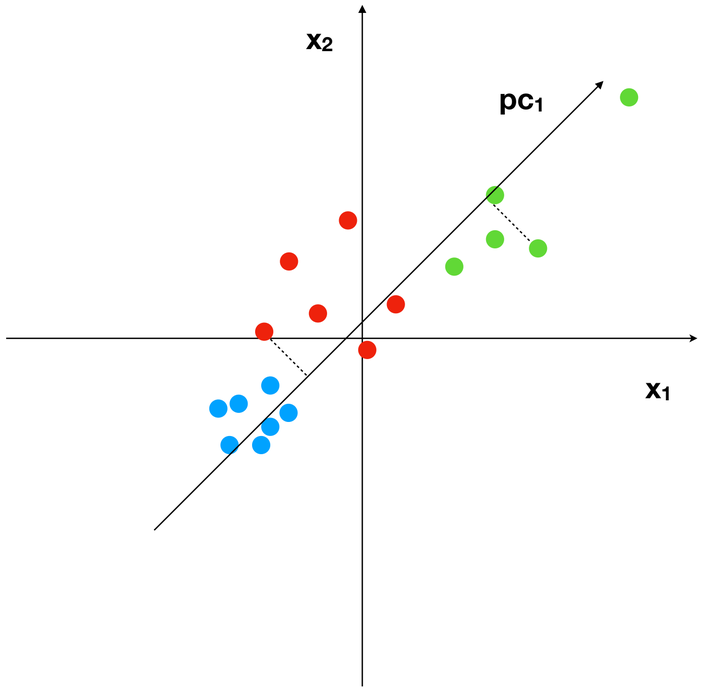


Shlens et al. 2014



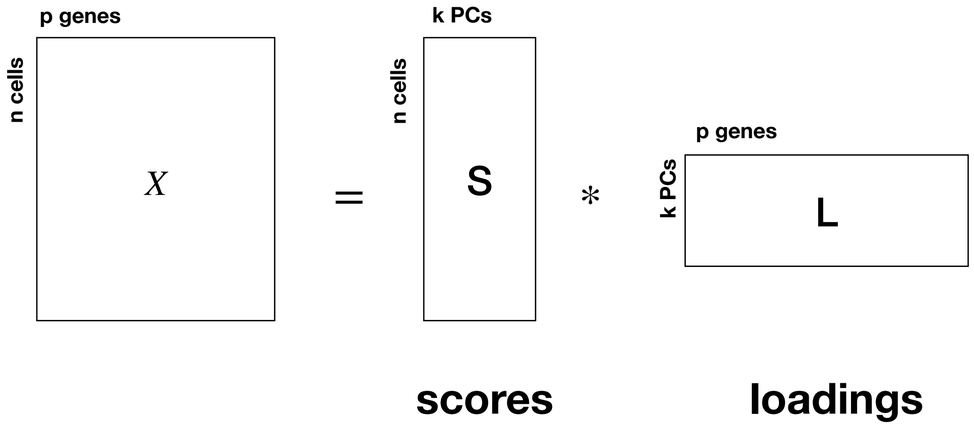
Principal component analysis (PCA)

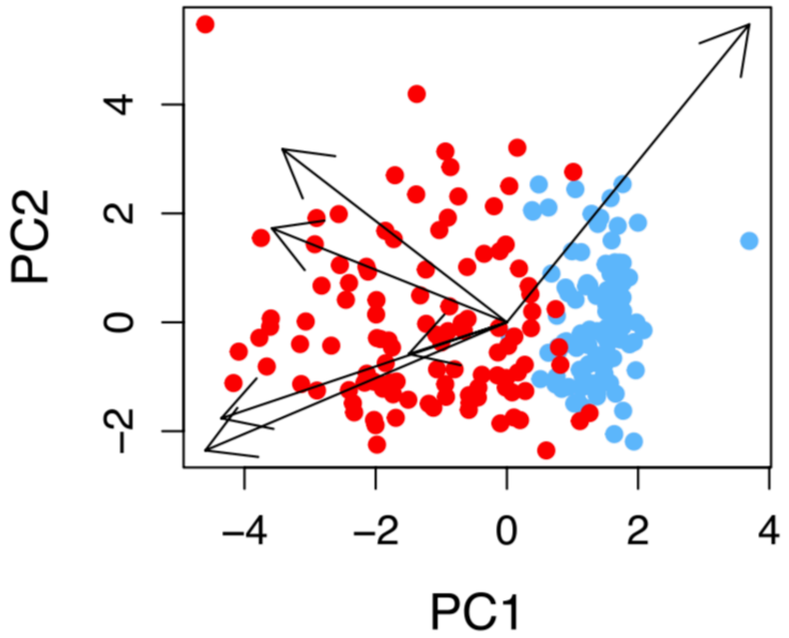




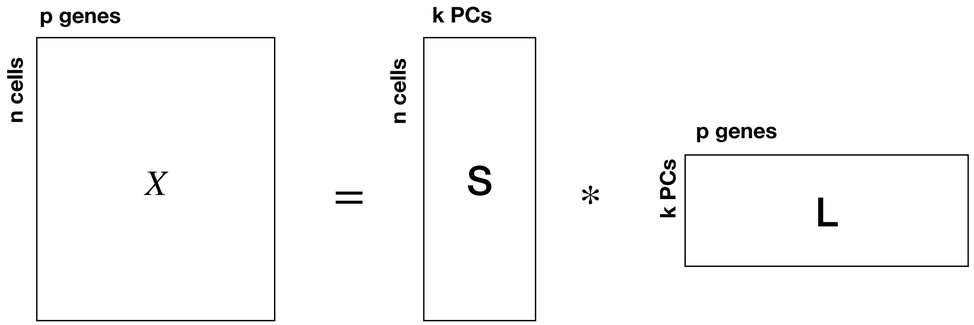
small data
MORE FROM LESS

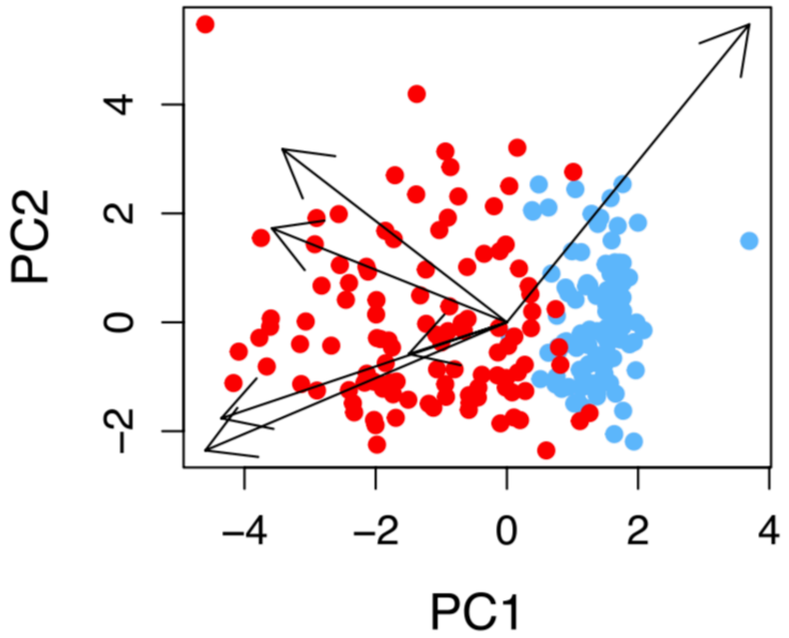
Principal component analysis (PCA)





Principal component analysis (PCA)

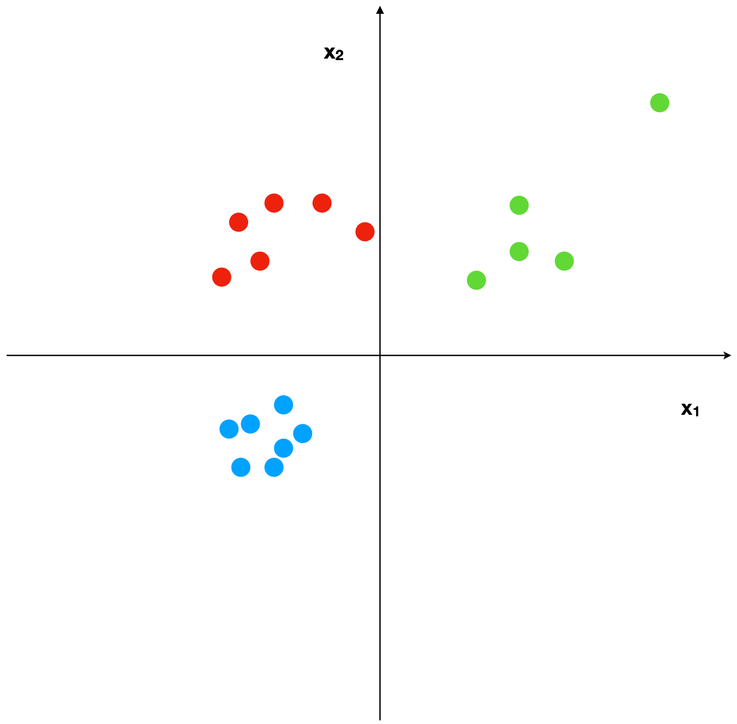




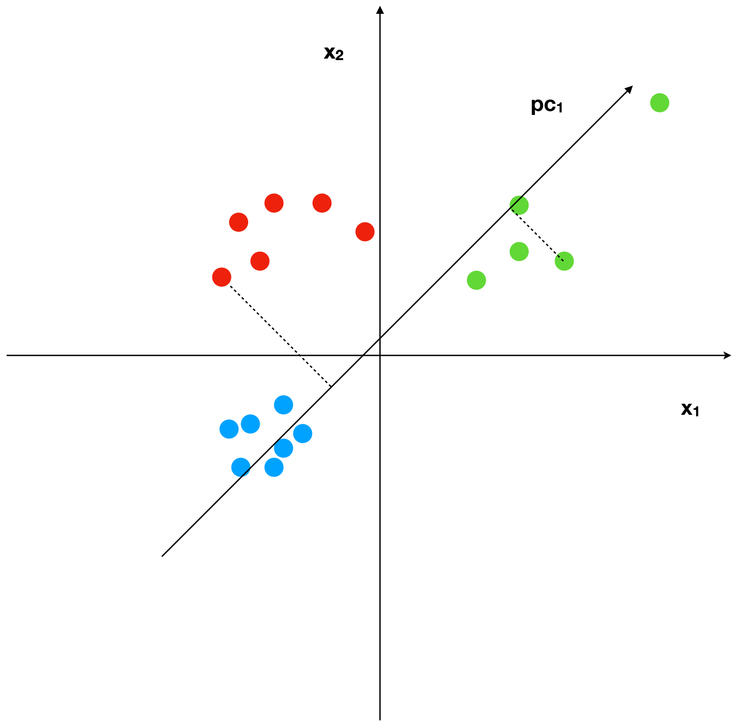
$$\min_{\mu,\lambda_1,\ldots,\lambda_n,\mathbf{V}_q} \sum_{i=1}^{n} \|x_i - \mu - \mathbf{V}_q\lambda_i\|^2$$

Principal components

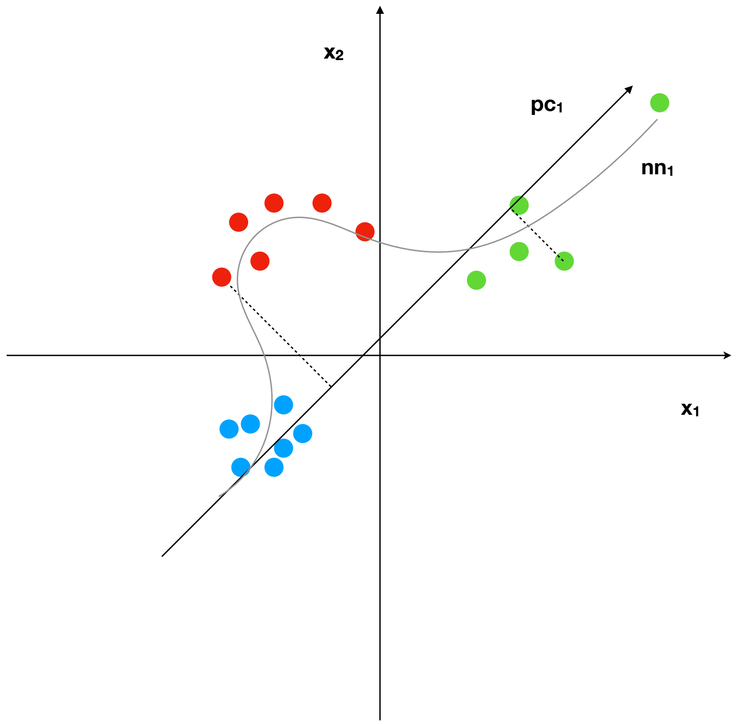
Variance - Bias Trade off



Variance - Bias Trade off

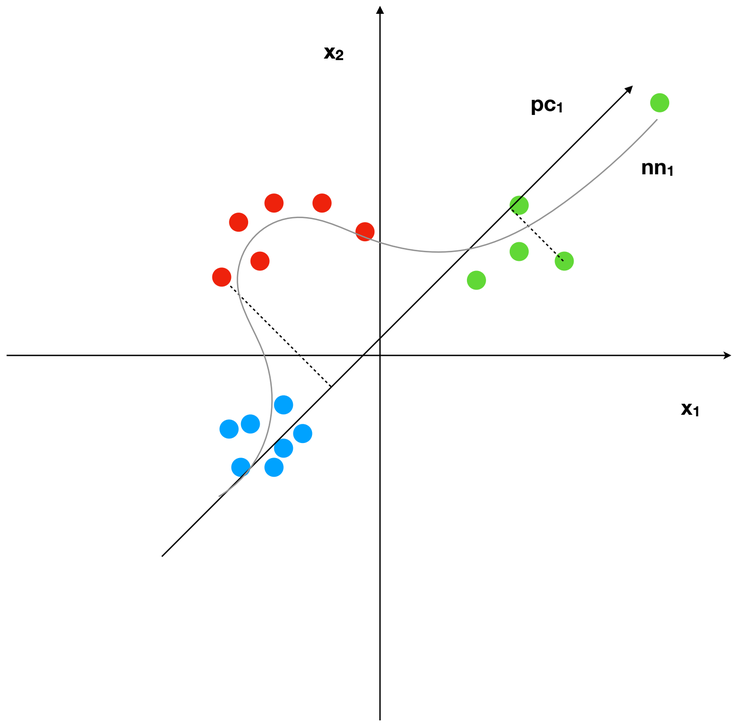


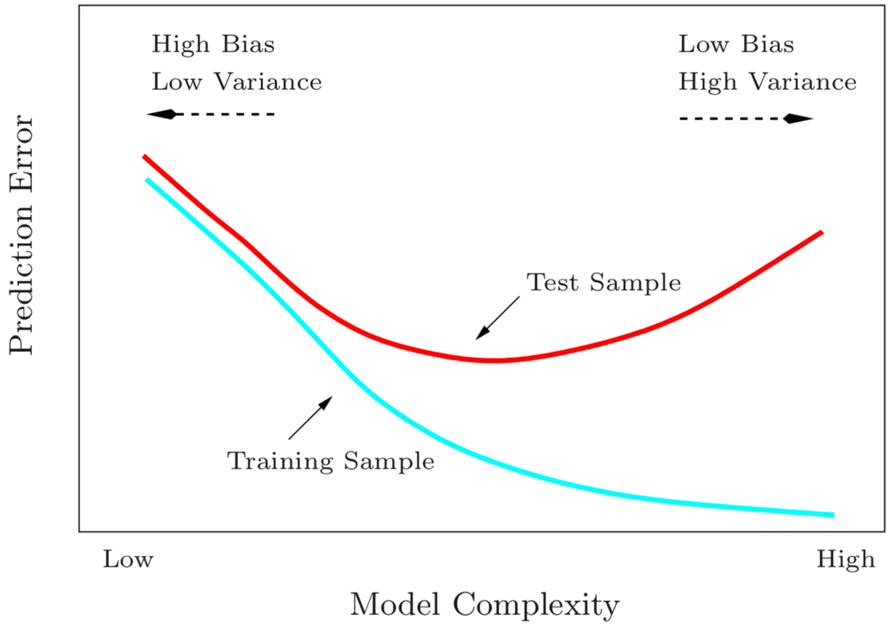
Variance - Bias Trade off



small data
MORE FROM LESS

Variance - Bias Trade off

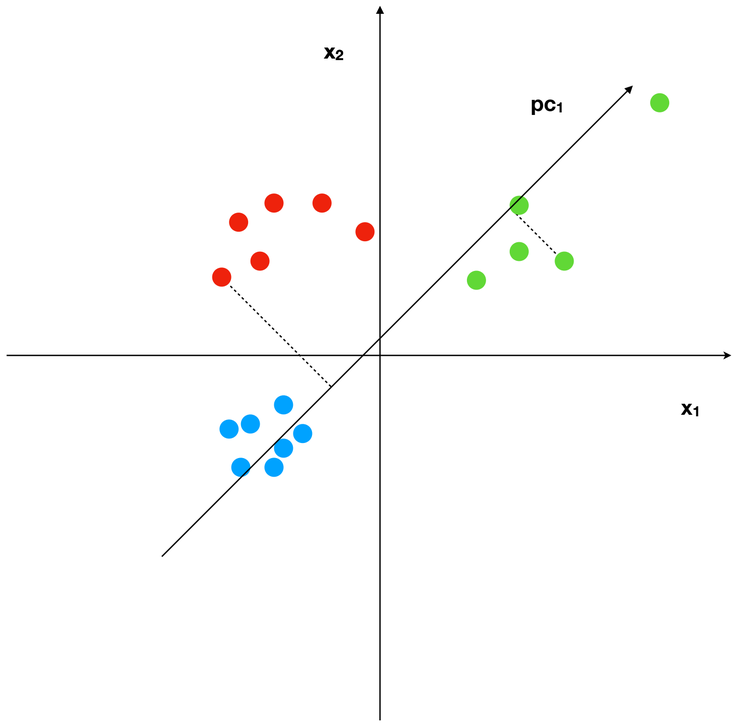




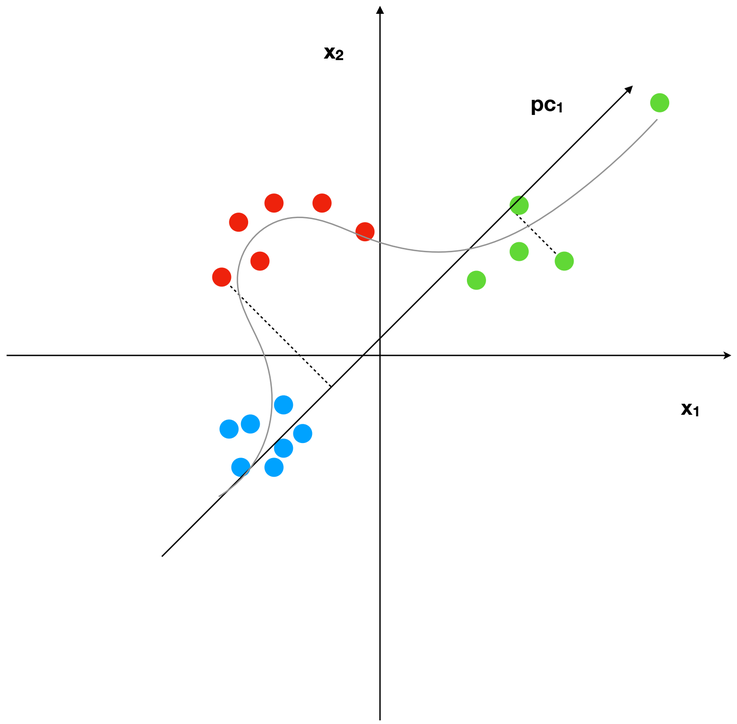
small data
MORE FROM LESS

Autoencoder

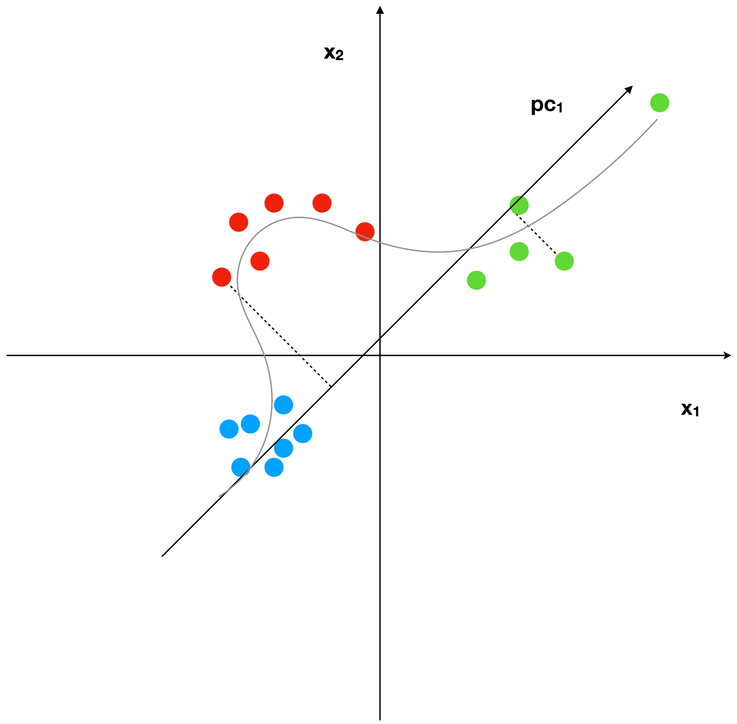
Autoencoder

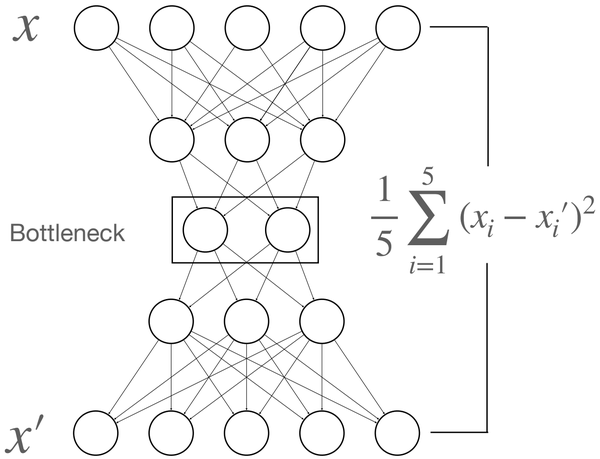


Autoencoder



Autoencoder

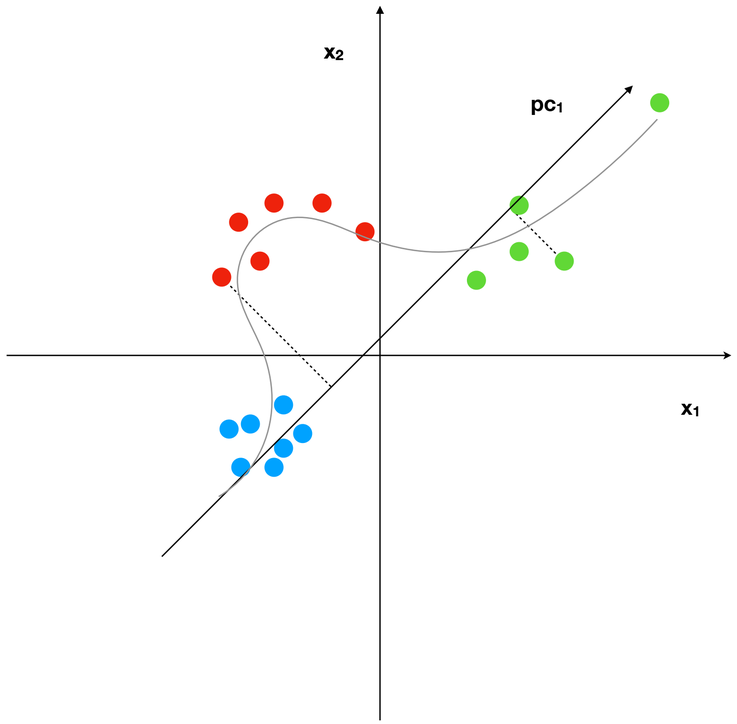


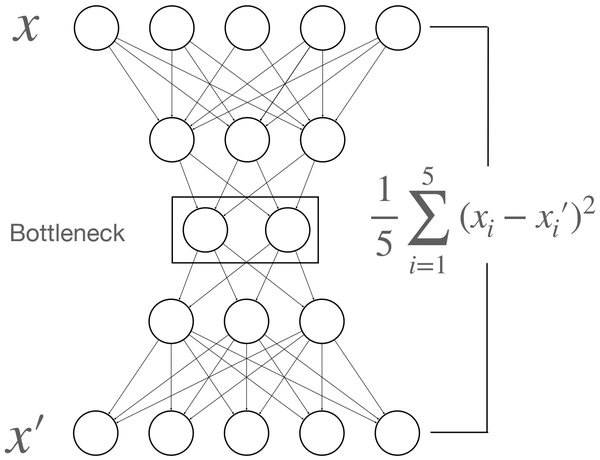


$x_1 =$	25	9	26	300	1
$x_1', \hat{x}_1 =$	20	4	20	310	1

small data
MORE FROM LESS

Autoencoder



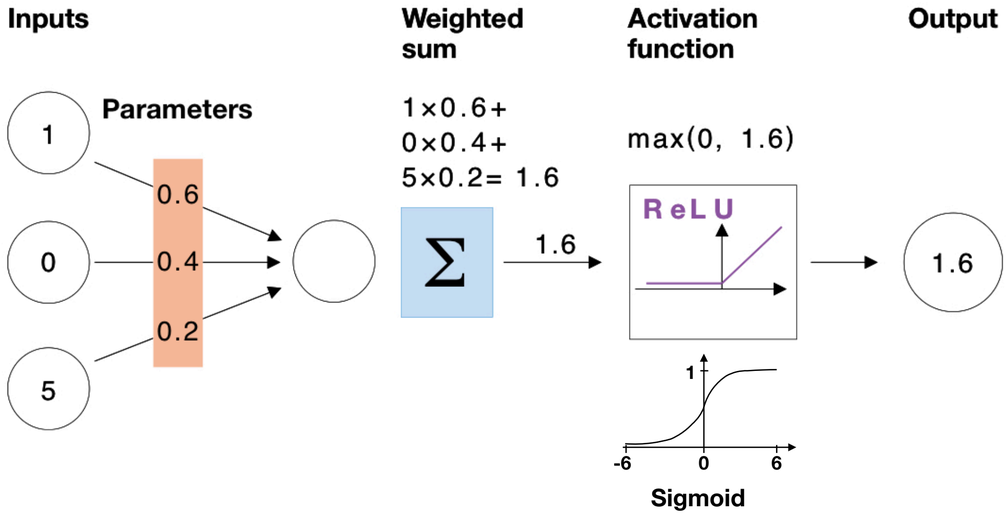


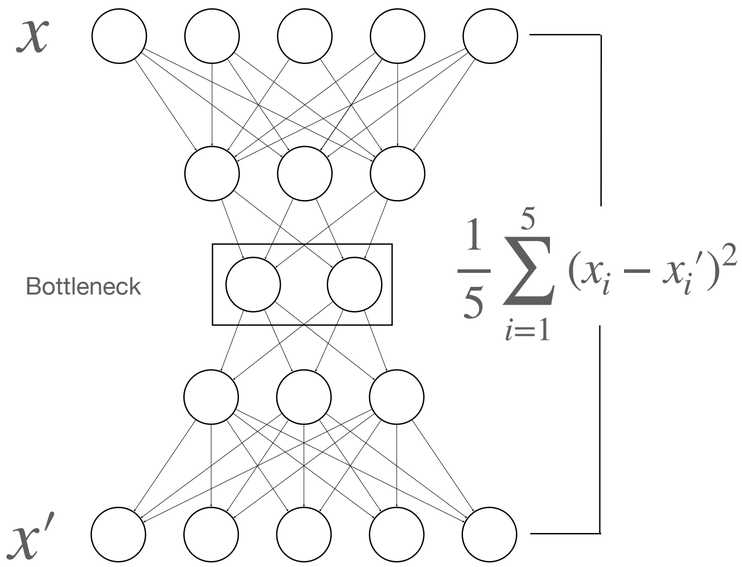
$x_1 =$	25	9	26	300	1
$x_1', \hat{x}_1 =$	20	4	20	310	1

$$L(\theta, \phi) := \mathbb{E}_{x \sim \mu_{ref}}\left[d(x, D_\theta(E_\phi(x)))\right]$$

$$\min_{\theta,\phi} L(\theta, \phi), \quad where \quad L(\theta, \phi) = \frac{1}{N}\sum_{i=1}^{N} \|x_i - D_\theta(E_\phi(x_i))\|_2^2$$

Learning non-linearities with deep neural networks

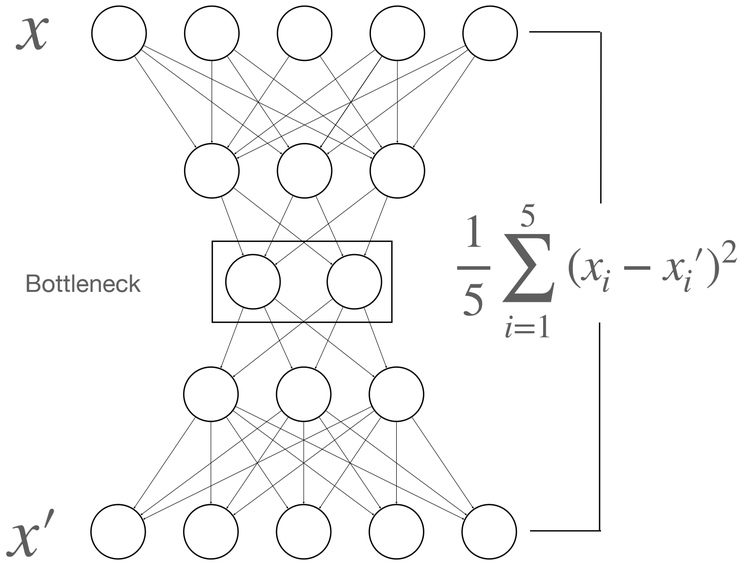


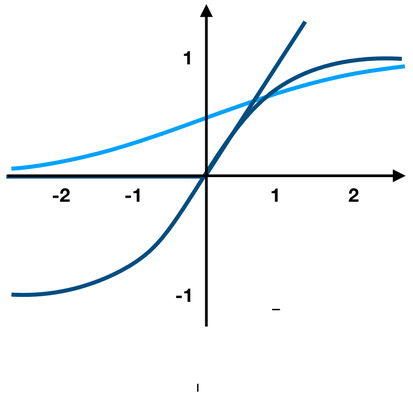


Adapted from Angermueller et al. 2016
by Martin Treppner

small data
MORE FROM LESS

NN mechanics and Hyperparameters

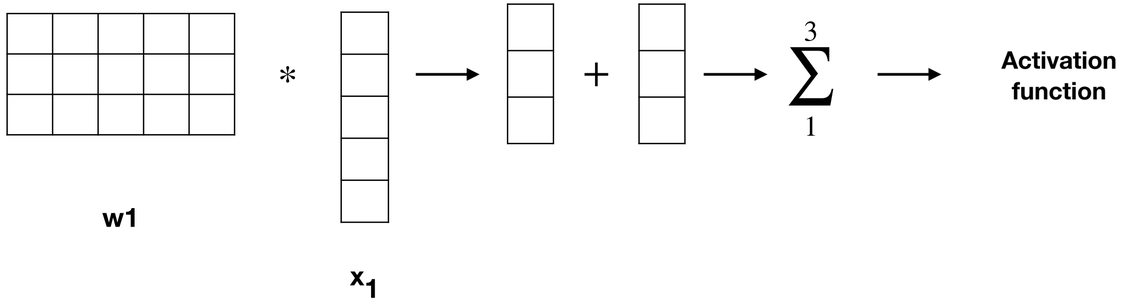




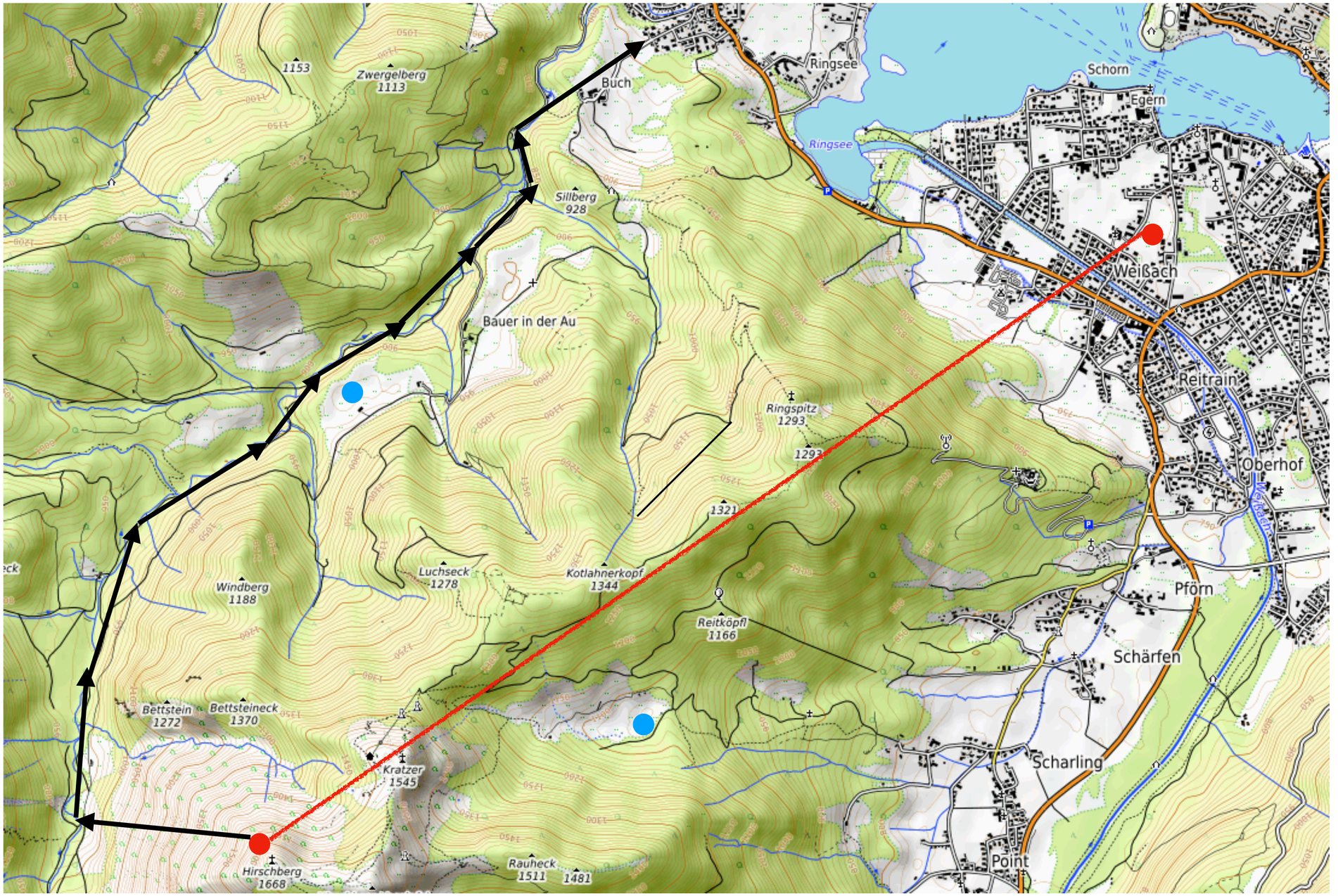
$$\sigma(x) = \frac{1}{1+e^{-x)}} \qquad \frac{d\sigma(x)}{dx} = \sigma(x) * 1 - \sigma(x)$$

$$tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})} \qquad \frac{dtanh(x)}{dx} = 1 - tanh^2(x)$$

$$ReLU(x) = max(0,x) \qquad \frac{ReLU(x)}{dx} = \begin{cases} 0 \textbf{ if } x < 0, \\ 1 \textbf{ if } x > 1 \end{cases}$$

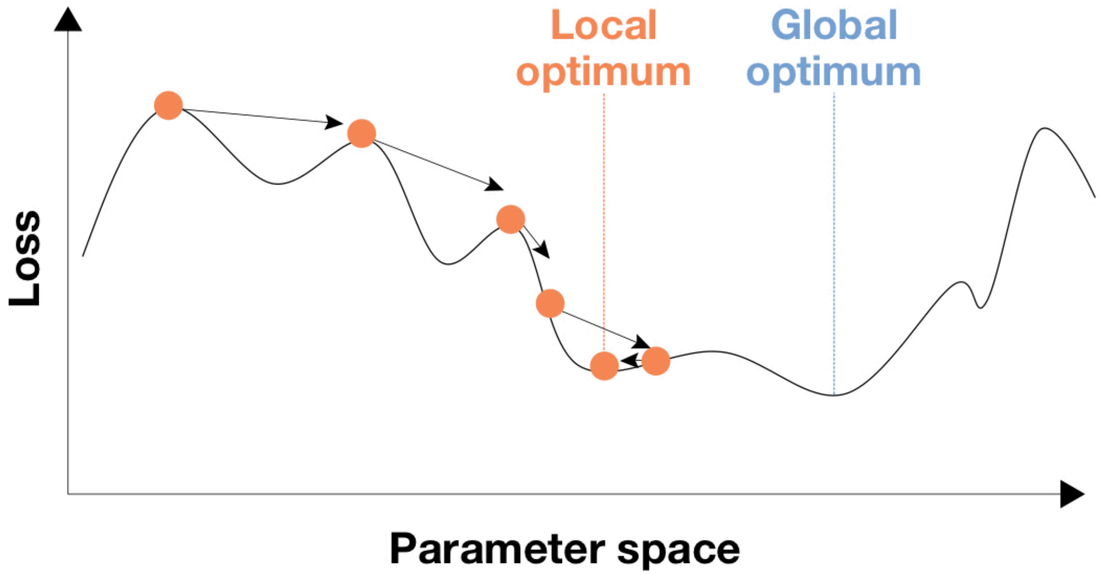


small data
MORE FROM LESS

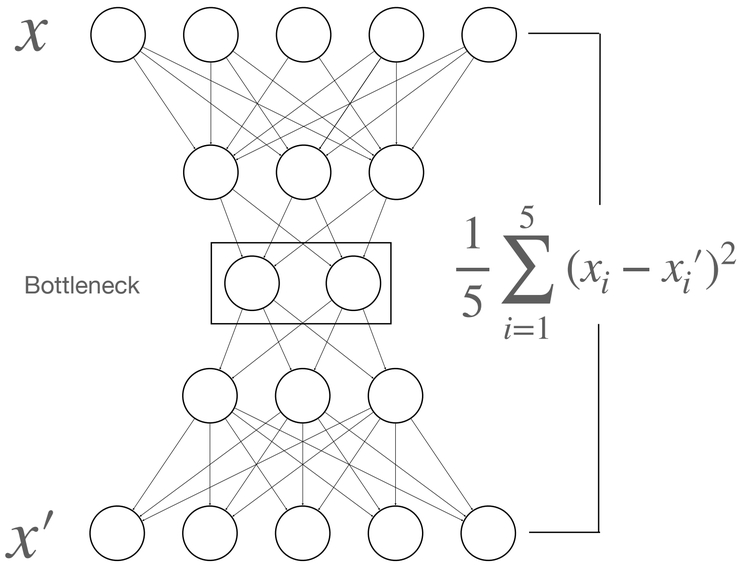


small data
MORE FROM LESS

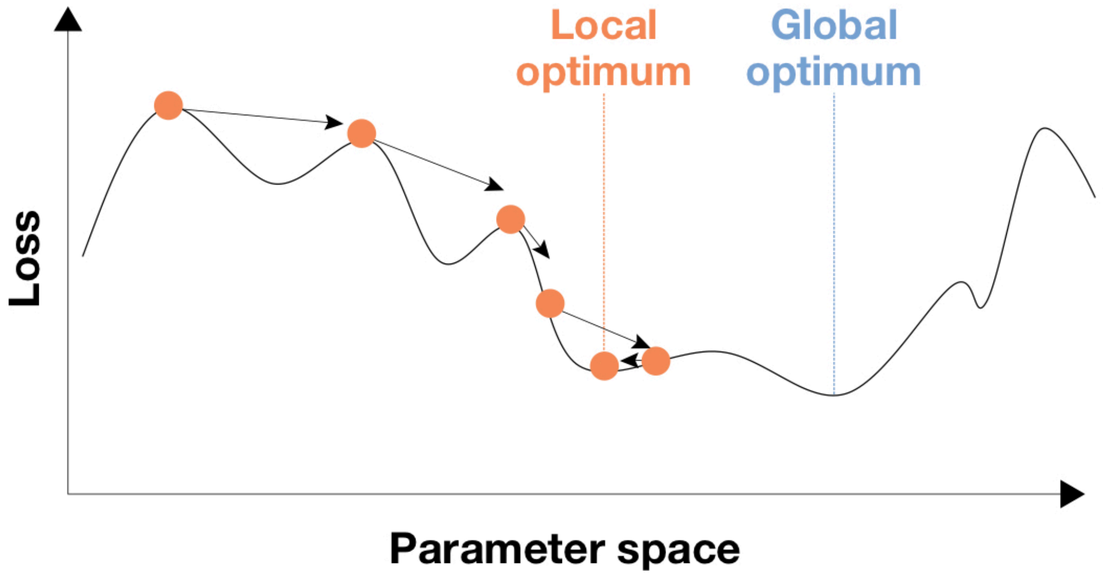
(Stochastic) gradient descent



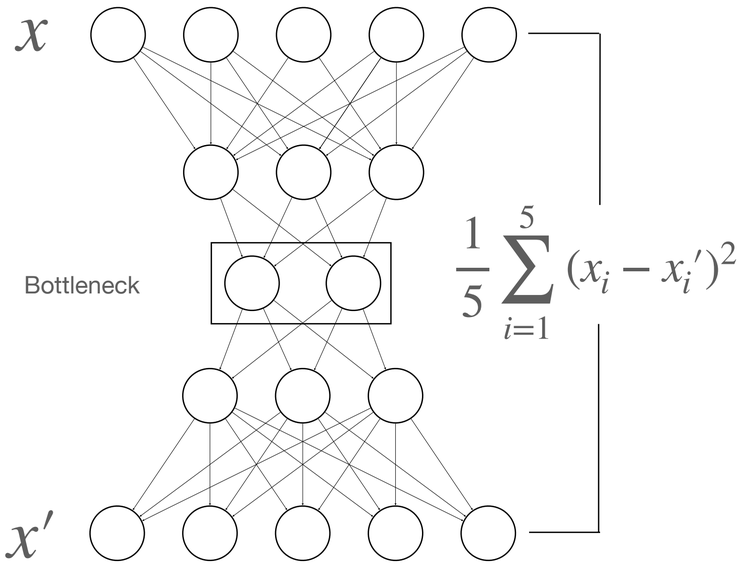
Adapted from Angermueller et al. 2016
by Martin Treppner



(Stochastic) gradient descent



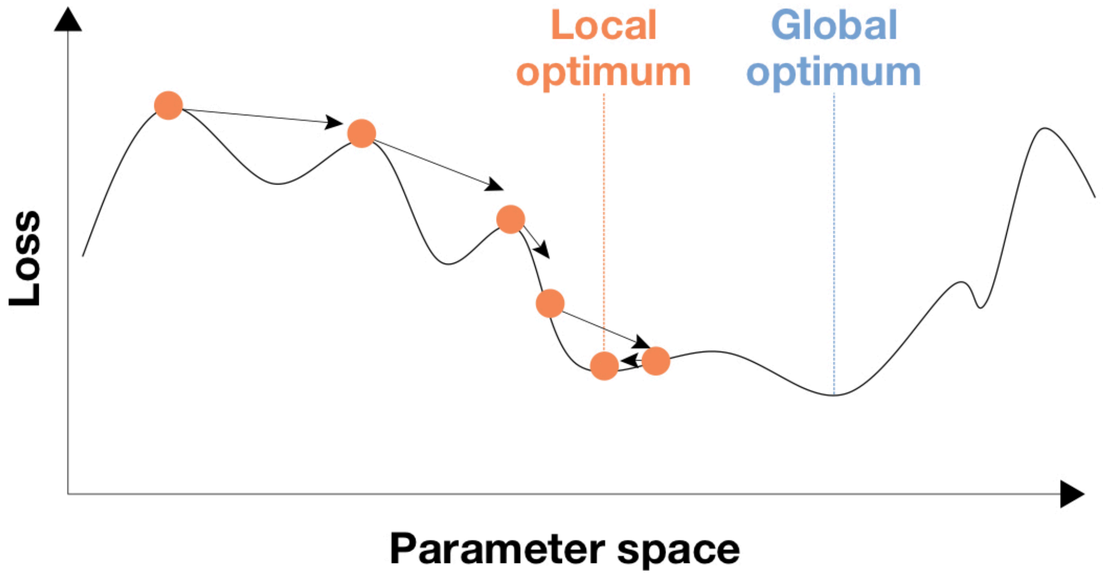
Adapted from Angermueller et al. 2016
by Martin Treppner



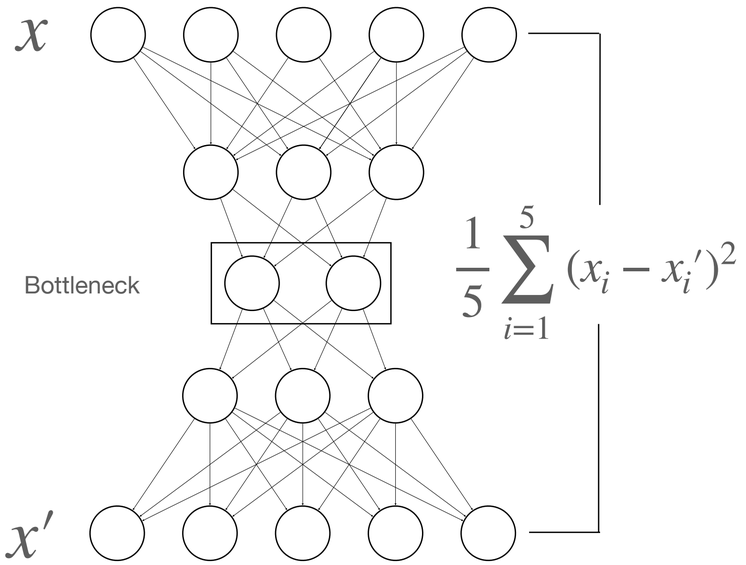
- set parameters (w) and learning rate η to random state
- Until approximate minimum is obtained do:
  - Shuffle n observations
  - Use batches of m observations where $1 \leq m < n$
  - For each batch do:
    - $w := w - \eta \nabla Q_i(w).$

small data
MORE FROM LESS

(Stochastic) gradient descent



Adapted from Angermueller et al. 2016
by Martin Treppner



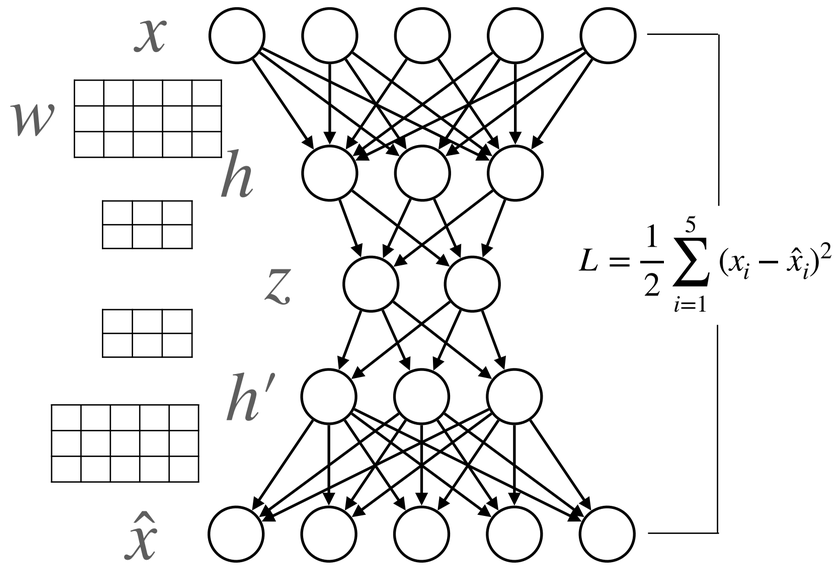
- set parameters (w) and learning rate η to random state
- Until approximate minimum is obtained do:
  - Shuffle n observations
  - Use batches of m observations where $1 \leq m < n$
  - For each batch do:
    - $w := w - \eta \nabla Q_i(w).$

- improved versions
  - AdaGrad
  - Adam

small data
MORE FROM LESS

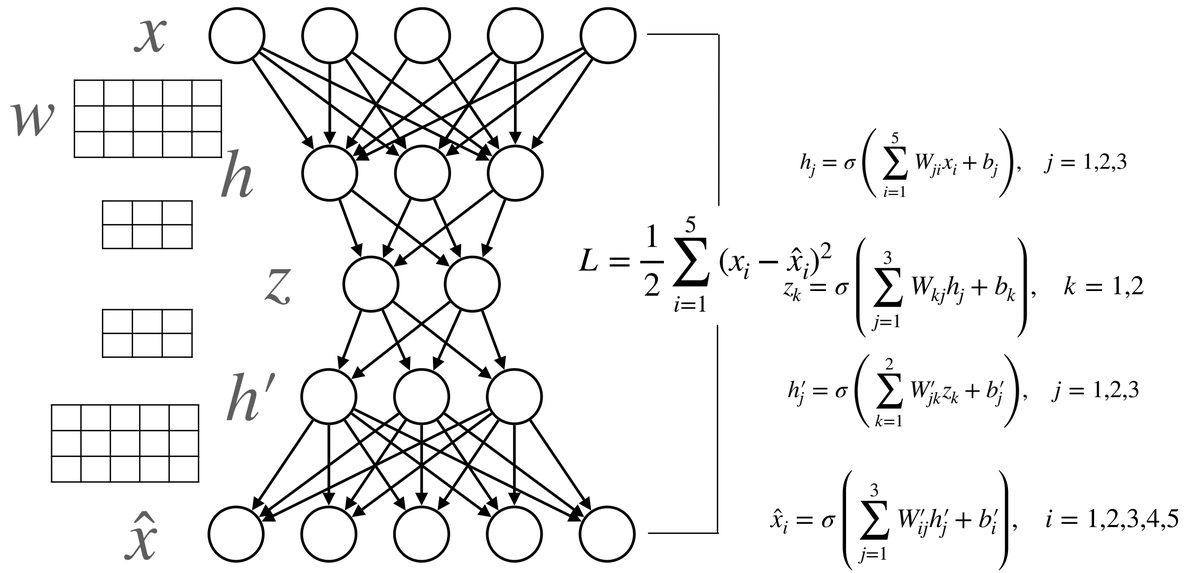
Backpropagation (of errors)

- Introduced by Rumelhart et al. (1986)
- Efficient way to compute the gradients for each NN parameter
- Forward pass:
  - Computing values
- Backward pass:
  - Computing gradients
  - Accumulating gradients
- Updating parameters with a fraction (learning-rate) of the gradients



Backpropagation (of errors)

- Introduced by Rumelhart et al. (1986)
- Efficient way to compute the gradients for each NN parameter
- Forward pass:
  - Computing values
- Backward pass:
  - Computing gradients
  - Accumulating gradients
- Updating parameters with a fraction (learning-rate) of the gradients



$$h_j = \sigma\left(\sum_{i=1}^{5} W_{ji}x_i + b_j\right), \quad j = 1,2,3$$

$$L = \frac{1}{2}\sum_{i=1}^{5}(x_i - \hat{x}_i)^2$$

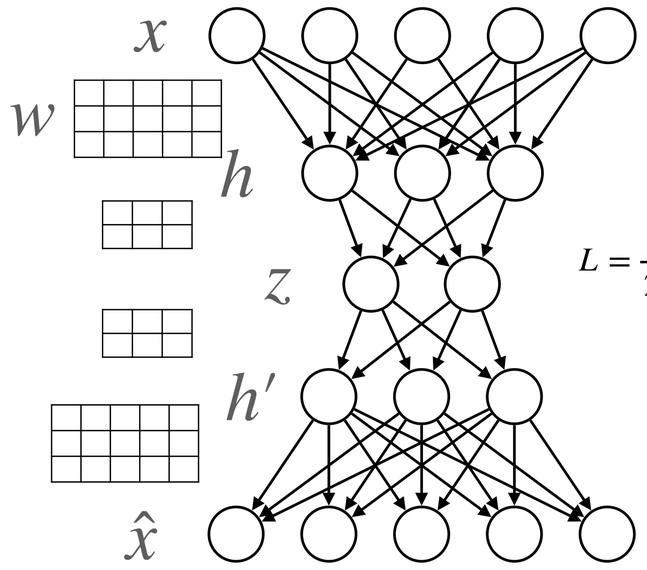
$$z_k = \sigma\left(\sum_{j=1}^{3} W_{kj}h_j + b_k\right), \quad k = 1,2$$

$$h'_j = \sigma\left(\sum_{k=1}^{2} W'_{jk}z_k + b'_j\right), \quad j = 1,2,3$$

$$\hat{x}_i = \sigma\left(\sum_{j=1}^{3} W'_{ij}h'_j + b'_i\right), \quad i = 1,2,3,4,5$$

Backpropagation (of errors)

- Introduced by Rumelhart et al. (1986)
- Efficient way to compute the gradients for each NN parameter
- Forward pass:
  - Computing values
- Backward pass:
  - Computing gradients
  - Accumulating gradients
- Updating parameters with a fraction (learning-rate) of the gradients



$$h_j = \sigma\left(\sum_{i=1}^{5} W_{ji}x_i + b_j\right), \quad j = 1,2,3$$

$$L = \frac{1}{2}\sum_{i=1}^{5}(x_i - \hat{x}_i)^2$$

$$z_k = \sigma\left(\sum_{j=1}^{3} W_{kj}h_j + b_k\right), \quad k = 1,2$$

$$h'_j = \sigma\left(\sum_{k=1}^{2} W'_{jk}z_k + b'_j\right), \quad j = 1,2,3$$

$$\hat{x}_i = \sigma\left(\sum_{j=1}^{3} W'_{ij}h'_j + b'_i\right), \quad i = 1,2,3,4,5$$

$$\frac{\partial L}{\partial \hat{x}_i} = \hat{x}_i - x_i$$

$$\frac{\partial L}{\partial W'_{ij}} = \frac{\partial L}{\partial \hat{x}_i} \cdot \sigma'(h'_j) \cdot h'_j$$

$$\frac{\partial L}{\partial h'_j} = \sum_{i=1}^{5} \frac{\partial L}{\partial \hat{x}_i} \cdot W'_{ij}$$

$$\frac{\partial L}{\partial W'_{jk}} = \frac{\partial L}{\partial h'_j} \cdot \sigma'(z_k) \cdot z_k$$

$$\frac{\partial L}{\partial z_k} = \sum_{j=1}^{3} \frac{\partial L}{\partial h'_j} \cdot W'_{jk}$$

$$\frac{\partial L}{\partial W_{kj}} = \frac{\partial L}{\partial z_k} \cdot \sigma'(h_j) \cdot h_j$$

$$\frac{\partial L}{\partial h_j} = \sum_{k=1}^{2} \frac{\partial L}{\partial z_k} \cdot W_{kj}$$

$$\frac{\partial L}{\partial W_{ji}} = \frac{\partial L}{\partial h_j} \cdot \sigma'(x_i) \cdot x_i$$

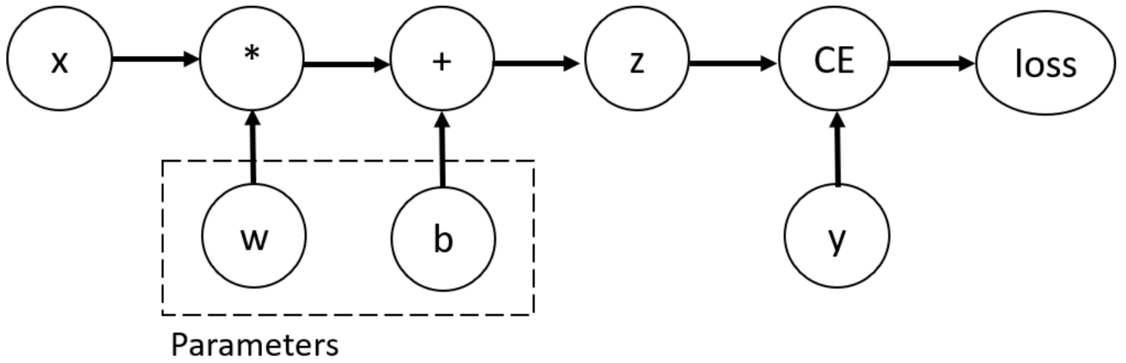
Automatic differentiation

Automatic differentiation

- NN are complex function
  - Composed of many simple functions
- Backpropagation requires gradients for NN parameters
- Automatic differentiation automatically computes
  - Values
  - Gradients
- Forward pass:
  - Computing values
- Backward pass:
  - Computing gradients
  - Accumulating gradients
- Multiple implementations
  - PyTorch (python)
  - TensorFlow (python)
  - Zygote (Julia)
  - No established AD framework for R
- AD frameworks are not specifically designed for NNs

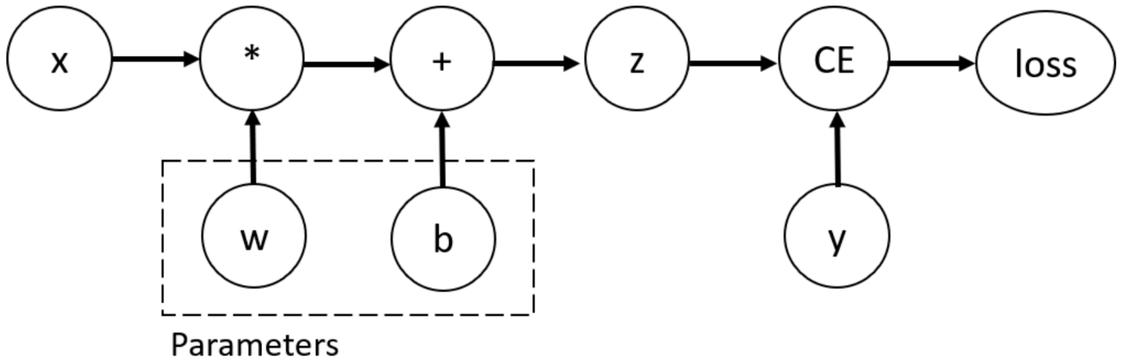
Automatic differentiation

- NN are complex function
  - Composed of many simple functions
- Backpropagation requires gradients for NN parameters
- Automatic differentiation automatically computes
  - Values
  - Gradients
- Forward pass:
  - Computing values
- Backward pass:
  - Computing gradients
  - Accumulating gradients
- Multiple implementations
  - PyTorch (python)
  - TensorFlow (python)
  - Zygote (Julia)
  - No established AD framework for R
- AD frameworks are not specifically designed for NNs



Automatic differentiation

- NN are complex function
  - Composed of many simple functions
- Backpropagation requires gradients for NN parameters
- Automatic differentiation automatically computes
  - Values
  - Gradients
- Forward pass:
  - Computing values
- Backward pass:
  - Computing gradients
  - Accumulating gradients
- Multiple implementations
  - PyTorch (python)
  - TensorFlow (python)
  - Zygote (Julia)
  - No established AD framework for R
- AD frameworks are not specifically designed for NNs



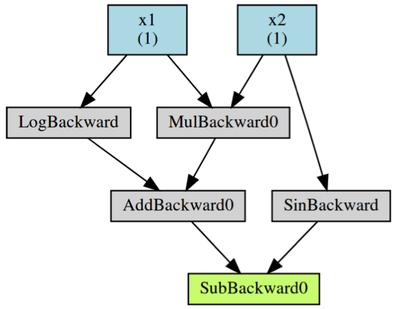

```python
import torch
def f(x1, x2):
    return torch.log(x1) + x1*x2 - torch.sin(x2)

x1 = torch.tensor([2.], requires_grad=True)
x2 = torch.tensor([5.], requires_grad=True)
y = f(x1, x2)
print("y:", y)

y.backward(torch.tensor([1.]))
print("x1 gradient:", x1._grad)
print("x2 gradient:", x2._grad)
```

```
y: tensor([11.6521], grad_fn=<SubBackward0>)
x1 gradient: tensor([5.5000])
x2 gradient: tensor([1.7163])
```

(a) Python code.



(b) computation graph.

small data
MORE FROM LESS

Automatic Differentiation II

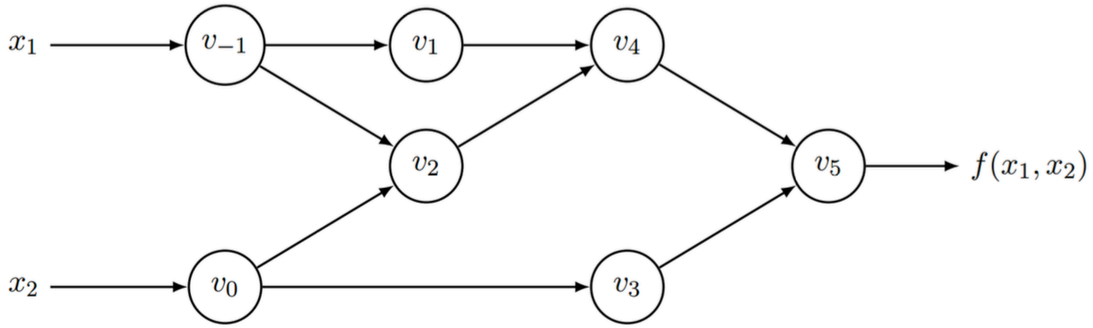


Figure 2: Computation graph for $y = f(x_1, x_2) = \ln(x_1) + x_1 x_2 - \sin(x_2)$.

Forward Primal Trace		Reverse Adjoint Trace		
$v_{-1} = x_1$	$= 2$	$x_1' = v_{-1}'$		$= 5.5$
$v_0 = x_2$	$= 5$	$x_2' = v_0'$		$= 1.716$
$v_1 = \ln(v_{-1})$	$= \ln(2)$	$v_{-1}' = v_{-1}' + v_1' \frac{\partial v_1}{\partial v_{-1}}$	$= v_{-1}' + v_1'/v_{-1}$	$= 5.5$
$v_2 = v_{-1} \times v_0$	$= 2 \times 5$	$v_0' = v_0' + v_2' \frac{\partial v_2}{\partial v_0}$	$= v_0' + v_2' \times v_{-1}$	$= 1.716$
		$v_{-1}' = v_2' \frac{\partial v_2}{\partial v_{-1}}$	$= v_2' \times v_0$	$= 5$
$v_3 = \sin(v_0)$	$= \sin(5)$	$v_0' = v_3' \frac{\partial v_3}{\partial v_0}$	$= v'3 \times \cos(v_0)$	$= -0.284$
$v_4 = v_1 + v_2$	$= 0.693 + 10$	$v_2' = v_4' \frac{\partial v_4}{\partial v_2}$	$= v_4' \times 1$	$= 1$
		$v_1' = v_4' \frac{\partial v_4}{\partial v_1}$	$= v_4' \times 1$	$= 1$
$v_5 = v_4 - v_3$	$= 10.693 + 0.959$	$v_3' = v_5' \frac{\partial v_5}{\partial v_3}$	$= v_5' \times (-1)$	$= -1$
		$v_4' = v_5' \frac{\partial v_5}{\partial v_4}$	$= v_5' \times 1$	$= 1$
$y = v_5$	$= 11.652$	$v_5' = y'$	$= 1$	

Table 1: Evaluation trace of $y = f(x_1, x_2) = \ln(x_1) + x_1 x_2 - \sin(x_2)$ with $x_1 = 2$ and $x_2 = 5$ using revers mode AD.

Practical 1: getting familiar with training (deep models) (30 min)

- First steps in python
- PCA via eigenvalue decomposition
- PCA via stochastic gradient descent
- Training Autoencoder
  - Combining layers
  - Non-linear activation functions (5)

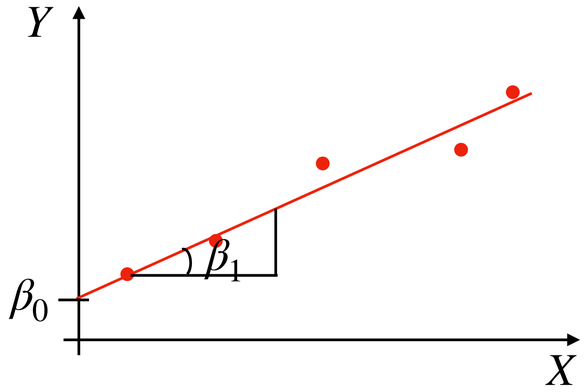
Part 2: deep generative approaches
Lecture 30 min, practicals 45 min

Discriminative vs. Generative Models

Discriminative vs. Generative Models

$$P(Y \mid X = x)$$

$$Y = \beta_0 + \beta_1 x_1 + \epsilon$$



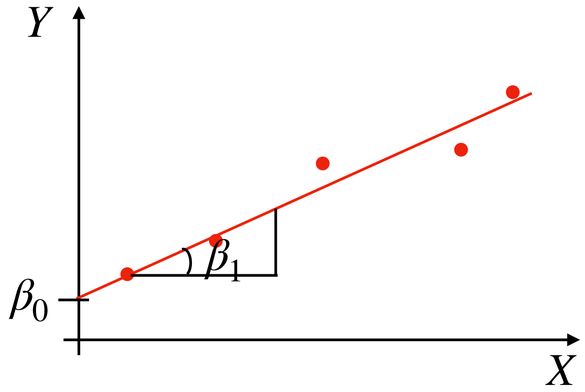
Likelihood $$L(\beta_0, \beta_1, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - (\beta_0 + \beta_1 x_i))^2}{2\sigma^2}\right)$$

$$\textbf{MSE}(\beta_0, \beta_1) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - (\beta_0 + \beta_1 x_i)\right)^2$$

Discriminative vs. Generative Models

$$P(Y \mid X = x)$$

$$Y = \beta_0 + \beta_1 x_1 + \epsilon$$



Likelihood $$L(\beta_0, \beta_1, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - (\beta_0 + \beta_1 x_i))^2}{2\sigma^2}\right)$$

$$\textbf{MSE}(\beta_0, \beta_1) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - (\beta_0 + \beta_1 x_i)\right)^2$$

$$P(X, Y)$$

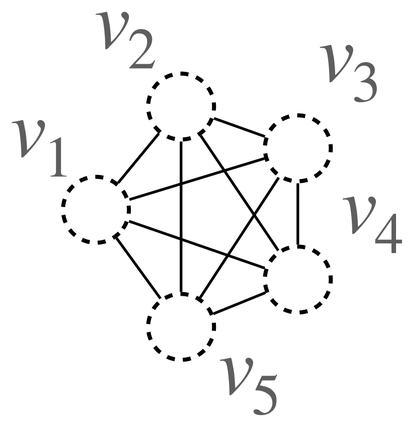
The simplest bivariate generative model: Log Linear Model

	$x = 0$	$x = 1$
$y = 0$		
$y = 1$		

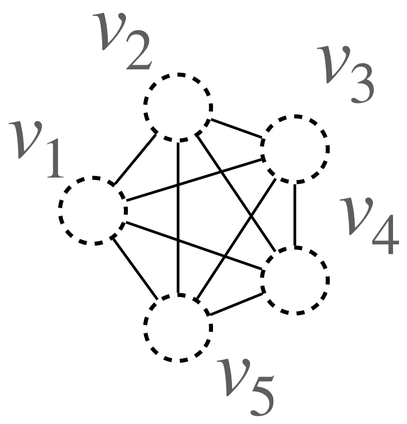
$$log(\mu_{i,j}) = \lambda + \lambda_i^X + \lambda_j^Y + \lambda_{ij}^{XY}$$

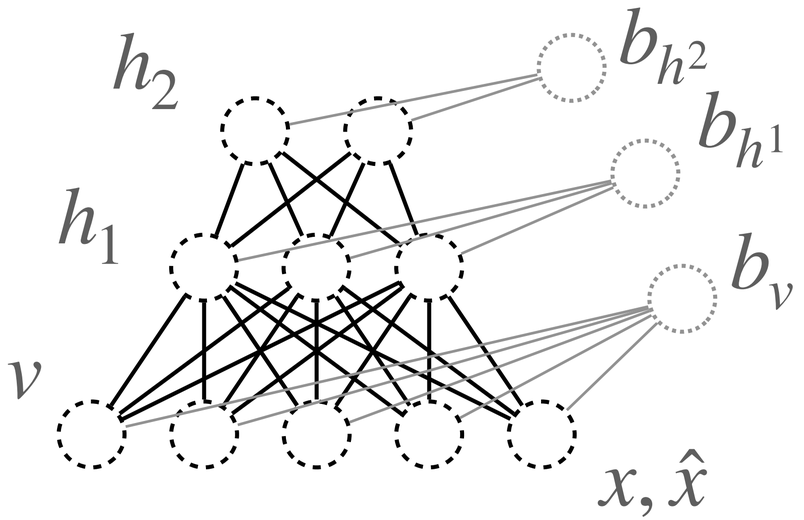
Deep Boltzmann machine

Deep Boltzmann machine



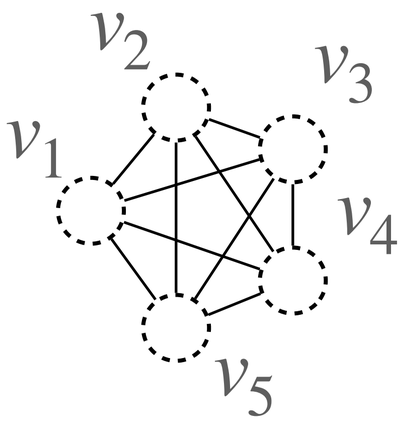
Deep Boltzmann machine

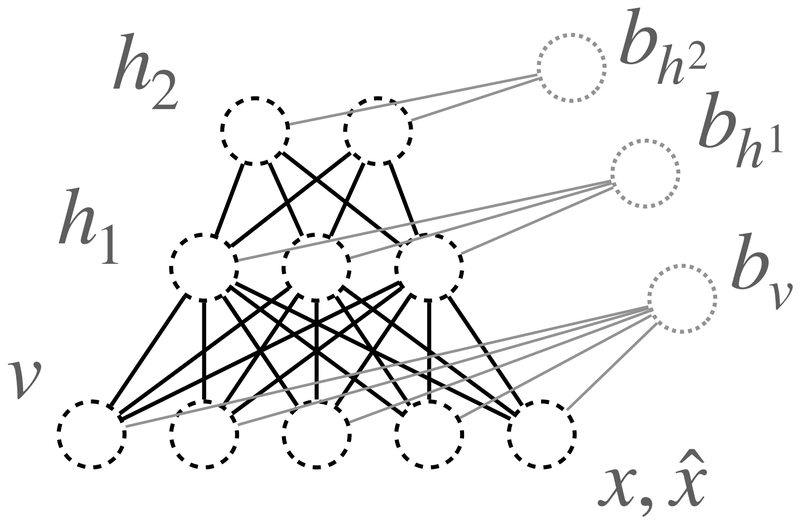


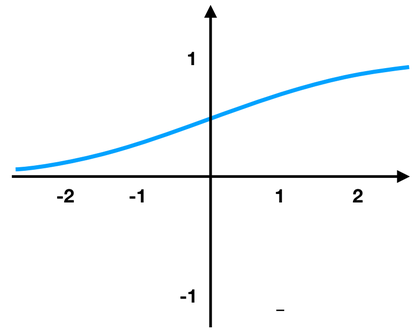


small data
MORE FROM LESS

Deep Boltzmann machine



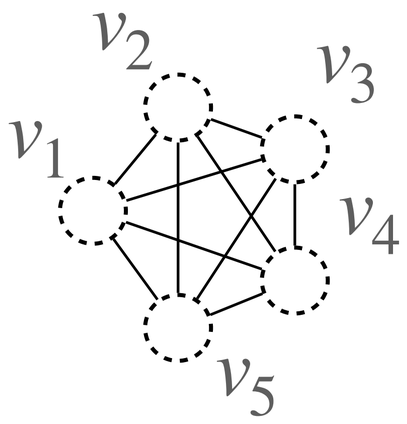


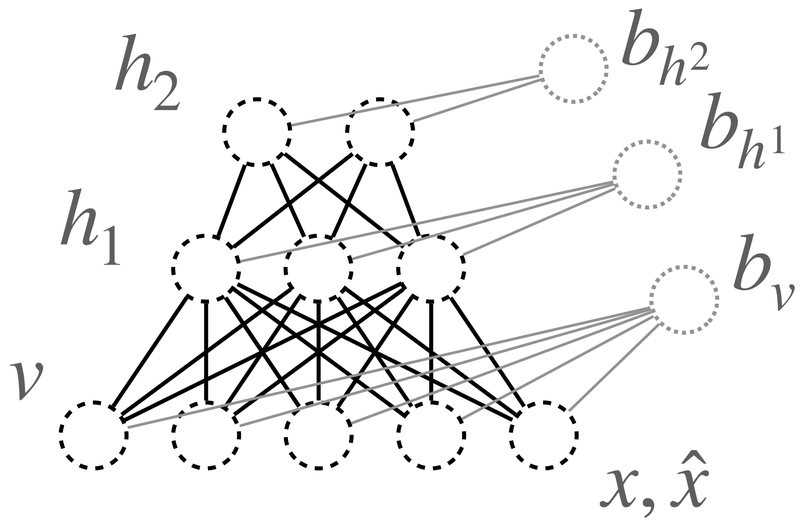


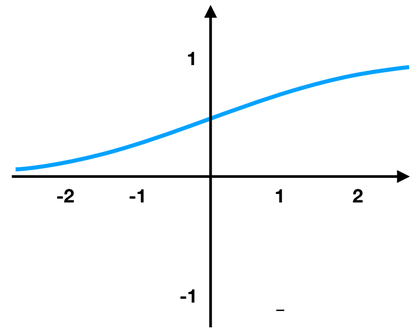
$$\sigma(x) = \frac{1}{1 + e^{-x)}}$$

small data
MORE FROM LESS

Deep Boltzmann machine







$$\sigma(x) = \frac{1}{1 + e^{-x)}}$$

- **Trained by Markov chain Monte Carlo (MCMC) approaches**

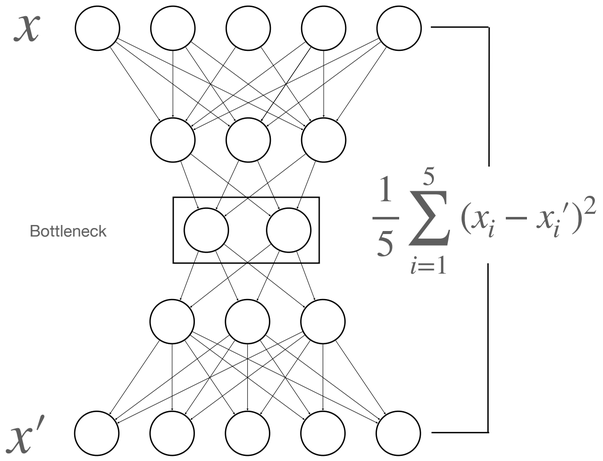
Joint probability: $P(v, h_1, h_2) = \frac{1}{Z} \exp(-E(v, h_1, h_2))$

Energy function: $E(v, h_1, h_2) = -v^\top W^{(1)} h_1 - h_1^\top W^{(2)} h_2 - b_v^\top v - b_{h_1}^\top h_1 - b_{h_2}^\top h_2$

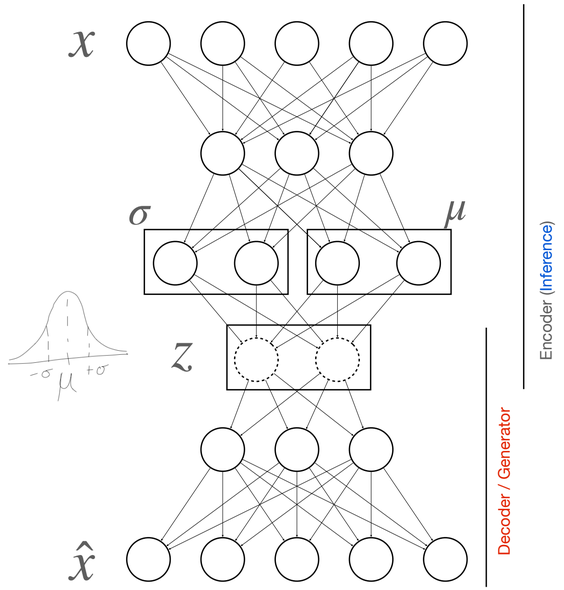
Objective function (Negative log-likelihood): $L(\theta) = -\sum_{i=1}^{N} \log P(v_i) = -\sum_{i=1}^{N} \log \sum_{h_1, h_2} \exp(-E(v_i, h_1, h_2))$

Variational Autoencoder (VAE)

Autoencoder



Variational Autoencoder



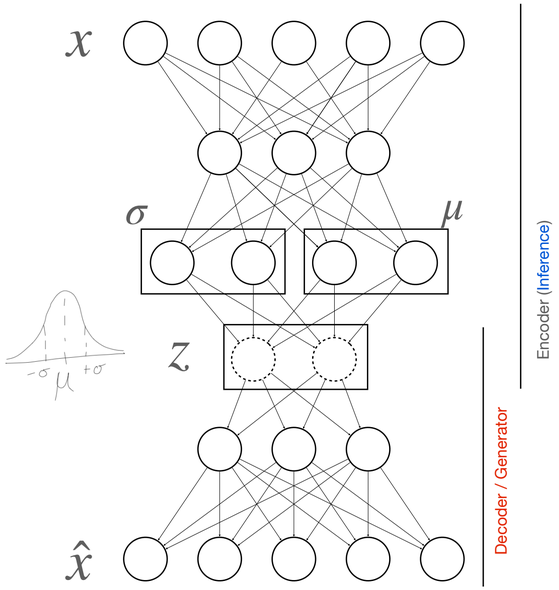
random

deterministic

$x_1 =$	25	9	26	300	1
$x_1', \hat{x}_1 =$	20	4	20	310	1

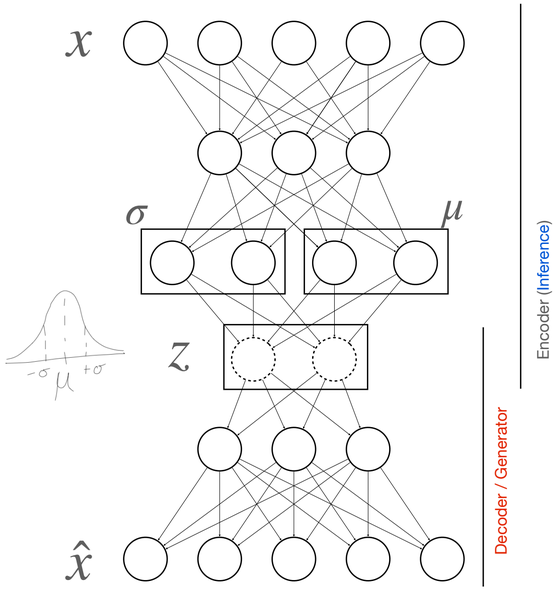
small data
MORE FROM LESS

“Extending” Autoencoders to a probabilistic random variable model



small data
MORE FROM LESS

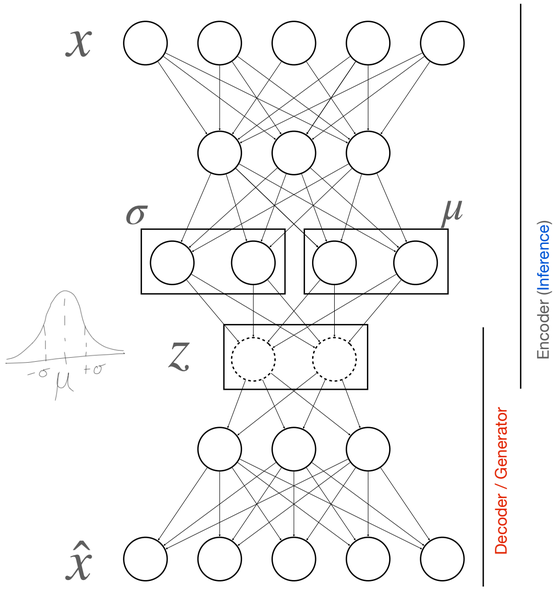
“Extending” Autoencoders to a probabilistic random variable model

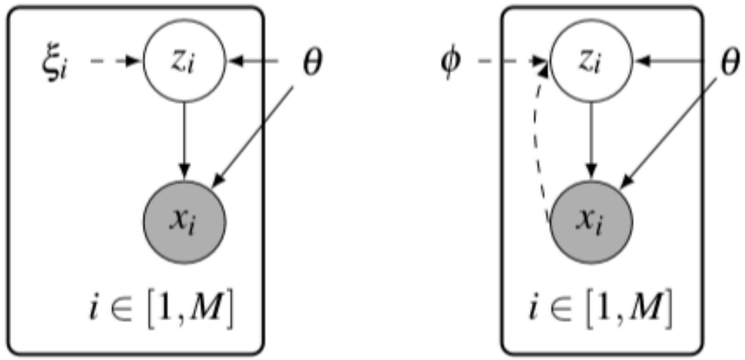


We want to learn the posterior distribution over the latent variables given the data.

$$p(z \mid x) = \frac{\overbrace{p(x \mid z)}^{\text{likelihood}} * \overbrace{p(z)}^{\text{prior}}}{\underbrace{p(x)}_{\text{evidence}}}$$

“Extending” Autoencoders to a probabilistic random variable model

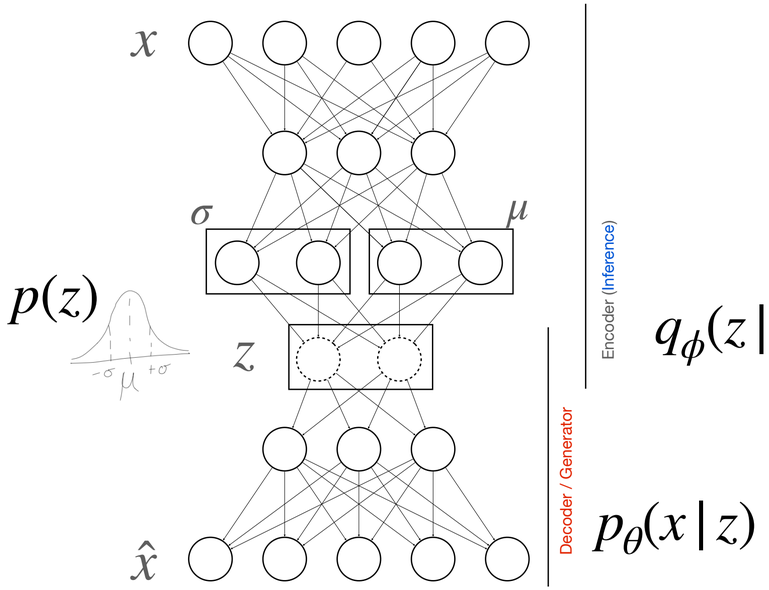




(a) SVI (b) VAE

Fig. 2. The graphical representation of stochastic variational inference (a) and the variational autoencoder (b). Dashed lines indicate variational approximations.

“Extending” Autoencoders to a probabilistic random variable model



$q_\phi(z \mid x)$

$p_\theta(x \mid z)$

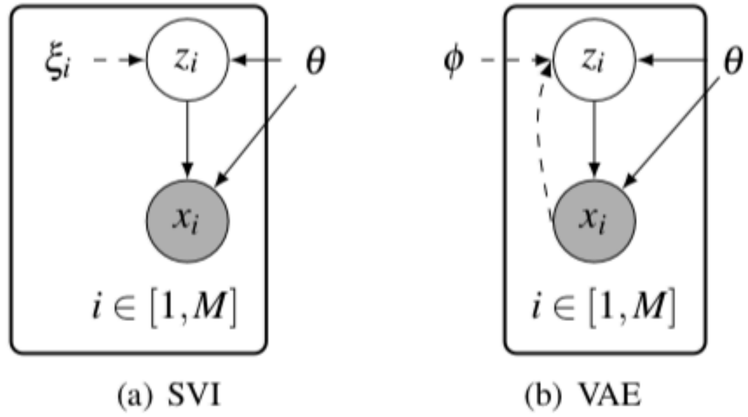


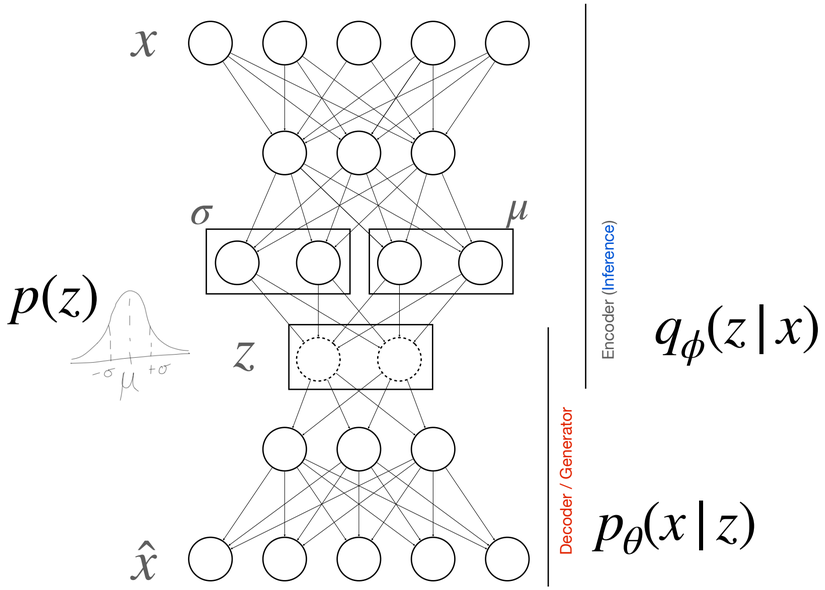
Fig. 2. The graphical representation of stochastic variational inference (a) and the variational autoencoder (b). Dashed lines indicate variational approximations.

Objective function: Evidence Lower BOund

$$log p_\theta(x_i) \geq \mathbb{E}_{q_\phi(z|x_i)}[log p_\theta(x_i \mid z)] - \mathbb{KL}(q_\phi(z \mid x_i) \mid\mid p(z))$$

reconstruction error **regularizer**

“Extending” Autoencoders to a probabilistic random variable model



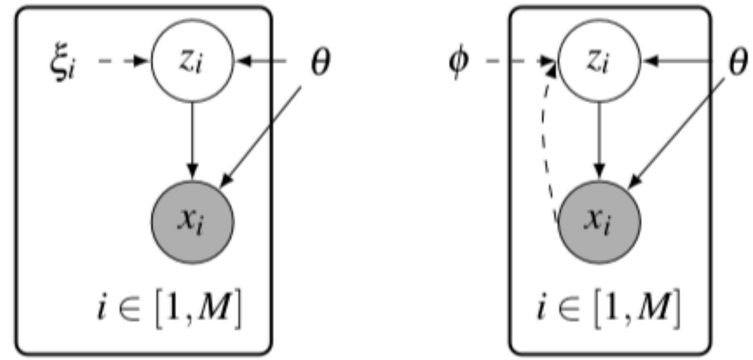
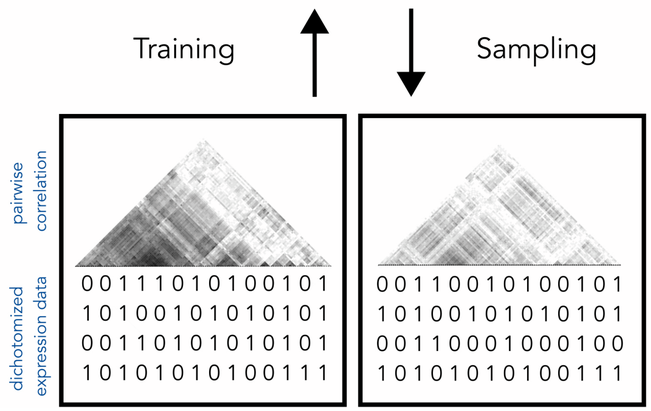


Fig. 2. The graphical representation of stochastic variational inference (a) and the variational autoencoder (b). Dashed lines indicate variational approximations.



Objective function: Evidence Lower BOund

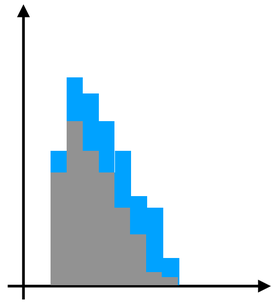
$$log p_\theta(x_i) \geq \mathbb{E}_{q_\phi(z|x_i)}[log p_\theta(x_i\,|\,z)] - \mathbb{KL}(q_\phi(z\,|\,x_i)\,||\,p(z))$$

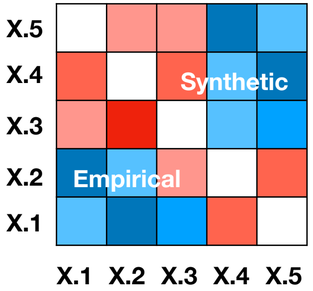
reconstruction error **regularizer**

small data
MORE FROM LESS

Evaluating VAEs by inspecting synthetic data

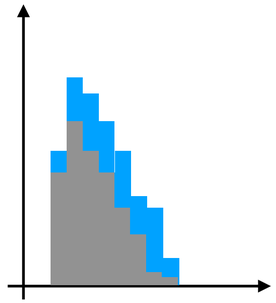
Evaluating VAEs by inspecting synthetic data

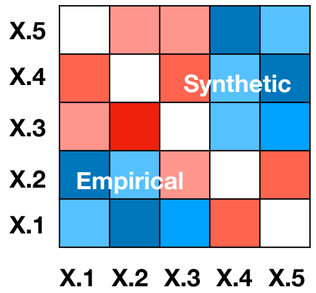


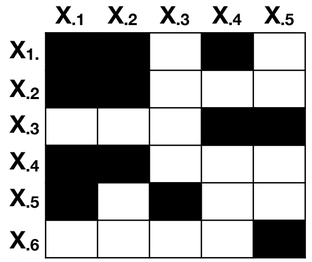


small data
MORE FROM LESS

Evaluating VAEs by inspecting synthetic data





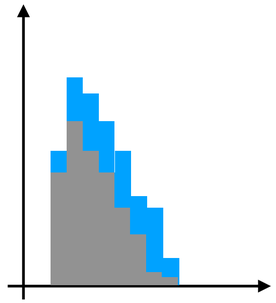


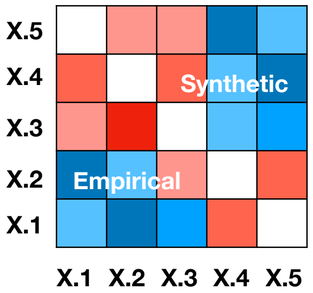
	$x_2 = 0$	$x_2 = 1$
$x_1 = 0$	$p(x_1 = 0 \mid x_2 = 0)$	$p(x_1 = 0 \mid x_2 = 1)$
$x_1 = 1$	$p(x_1 = 1 \mid x_2 = 0)$	$p(x_1 = 1 \mid x_2 = 1)$

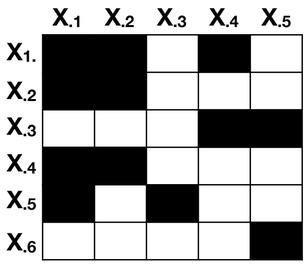
$$\frac{p(x_1 = 0 \mid x_2 = 0) * p(x_1 = 1 \mid x_2 = 1)}{p(x_1 = 0 \mid x_2 = 1) * p(x_1 = 1 \mid x_2 = 0)}$$

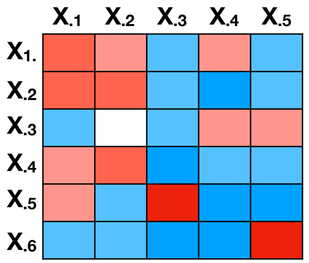
small data
MORE FROM LESS

Evaluating VAEs by inspecting synthetic data







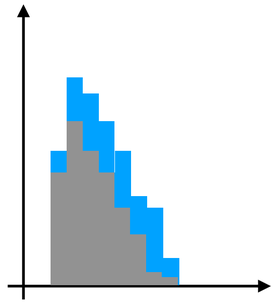


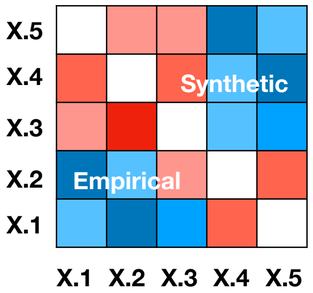
	$x_2 = 0$	$x_2 = 1$
$x_1 = 0$	$p(x_1 = 0 \mid x_2 = 0)$	$p(x_1 = 0 \mid x_2 = 1)$
$x_1 = 1$	$p(x_1 = 1 \mid x_2 = 0)$	$p(x_1 = 1 \mid x_2 = 1)$

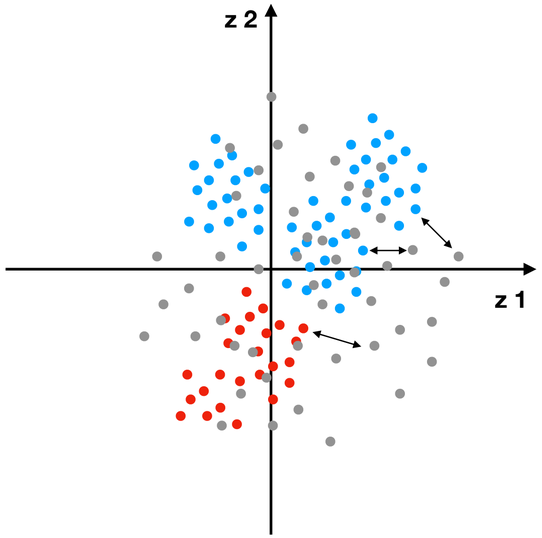
$$\frac{p(x_1 = 0 \mid x_2 = 0) * p(x_1 = 1 \mid x_2 = 1)}{p(x_1 = 0 \mid x_2 = 1) * p(x_1 = 1 \mid x_2 = 0)}$$

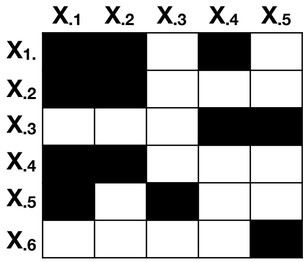
$$\textbf{Cov}(x_1, x_2) = \mathbb{E}[(x_1 - \mathbb{E}[x_1])(x_2 - \mathbb{E}[x_2])]$$

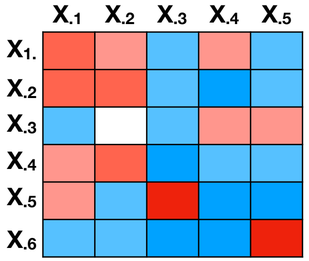
Evaluating VAEs by inspecting synthetic data











	$x_2 = 0$	$x_2 = 1$
$x_1 = 0$	$p(x_1 = 0 \mid x_2 = 0)$	$p(x_1 = 0 \mid x_2 = 1)$
$x_1 = 1$	$p(x_1 = 1 \mid x_2 = 0)$	$p(x_1 = 1 \mid x_2 = 1)$

$$\frac{p(x_1 = 0 \mid x_2 = 0) * p(x_1 = 1 \mid x_2 = 1)}{p(x_1 = 0 \mid x_2 = 1) * p(x_1 = 1 \mid x_2 = 0)}$$

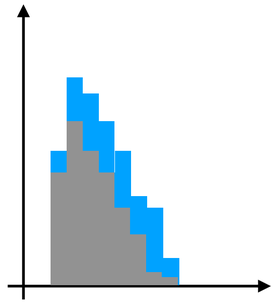
$$\mathbf{Cov}(x_1, x_2) = \mathbb{E}[(x_1 - \mathbb{E}[x_1])(x_2 - \mathbb{E}[x_2])]$$

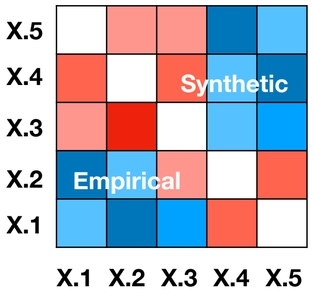
$$T_{m,n} = \frac{mn}{m+n}\left(\frac{2}{mn}\sum_{i,j}^{m,n}\varphi(\|\mathbf{X}_i - \hat{\mathbf{X}}_j\|^2) - \frac{1}{m^2}\sum_{i,j=1}^{m}\varphi(\|\mathbf{X}_i - \mathbf{X}_j\|^2) - \frac{1}{n^2}\sum_{i,j=1}^{n}\varphi(\|\hat{\mathbf{X}}_i - \hat{\mathbf{X}}_j\|^2)\right)$$

Cramér test / statistic for comparing two multivariate distributions

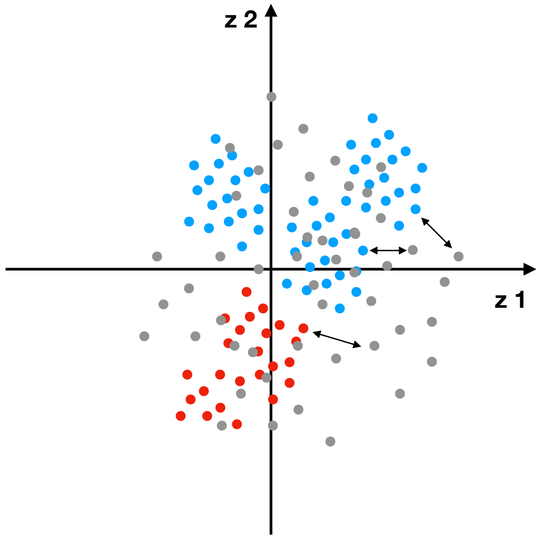
small data
MORE FROM LESS

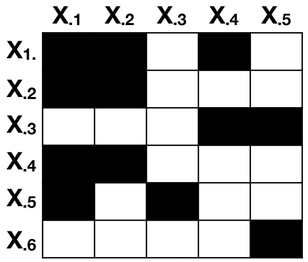
Evaluating VAEs by inspecting synthetic data

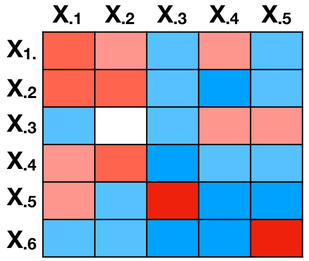




- Covariance
- Odds Ratios
- Similarity Real, Synthetic
- Marginal Distributions







	$x_2 = 0$	$x_2 = 1$
$x_1 = 0$	$p(x_1 = 0 \mid x_2 = 0)$	$p(x_1 = 0 \mid x_2 = 1)$
$x_1 = 1$	$p(x_1 = 1 \mid x_2 = 0)$	$p(x_1 = 1 \mid x_2 = 1)$

$$\frac{p(x_1 = 0 \mid x_2 = 0) * p(x_1 = 1 \mid x_2 = 1)}{p(x_1 = 0 \mid x_2 = 1) * p(x_1 = 1 \mid x_2 = 0)}$$

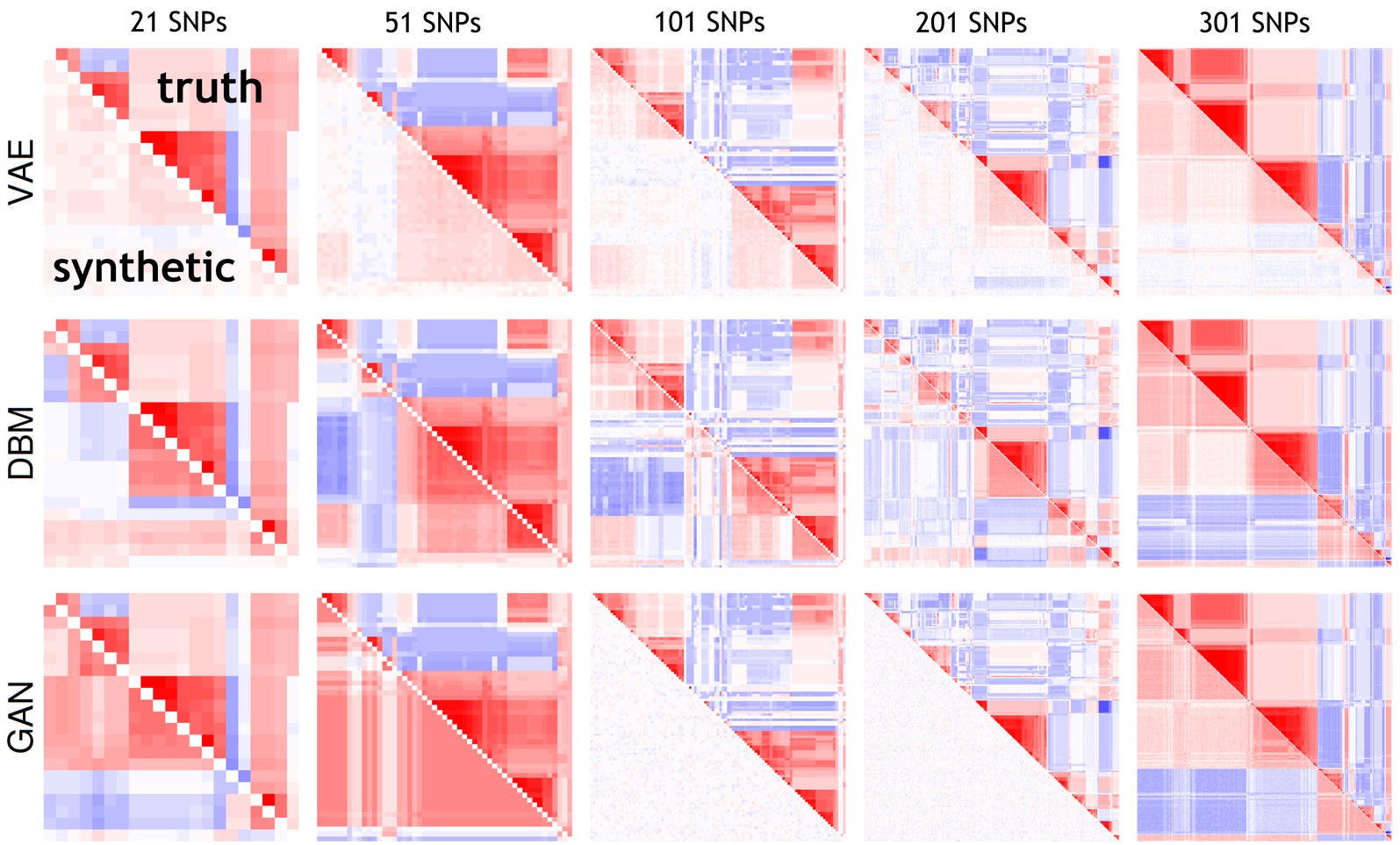
$$\mathbf{Cov}(x_1, x_2) = \mathbb{E}[(x_1 - \mathbb{E}[x_1])(x_2 - \mathbb{E}[x_2])]$$

$$T_{m,n} = \frac{mn}{m+n}\left(\frac{2}{mn}\sum_{i,j}^{m,n}\varphi(\|\mathbf{X}_i - \hat{\mathbf{X}}_j\|^2) - \frac{1}{m^2}\sum_{i,j=1}^{m}\varphi(\|\mathbf{X}_i - \mathbf{X}_j\|^2) - \frac{1}{n^2}\sum_{i,j=1}^{n}\varphi(\|\hat{\mathbf{X}}_i - \hat{\mathbf{X}}_j\|^2)\right)$$

Cramér test / statistic for comparing two multivariate distributions

small data
MORE FROM LESS

Effect of sample size and data



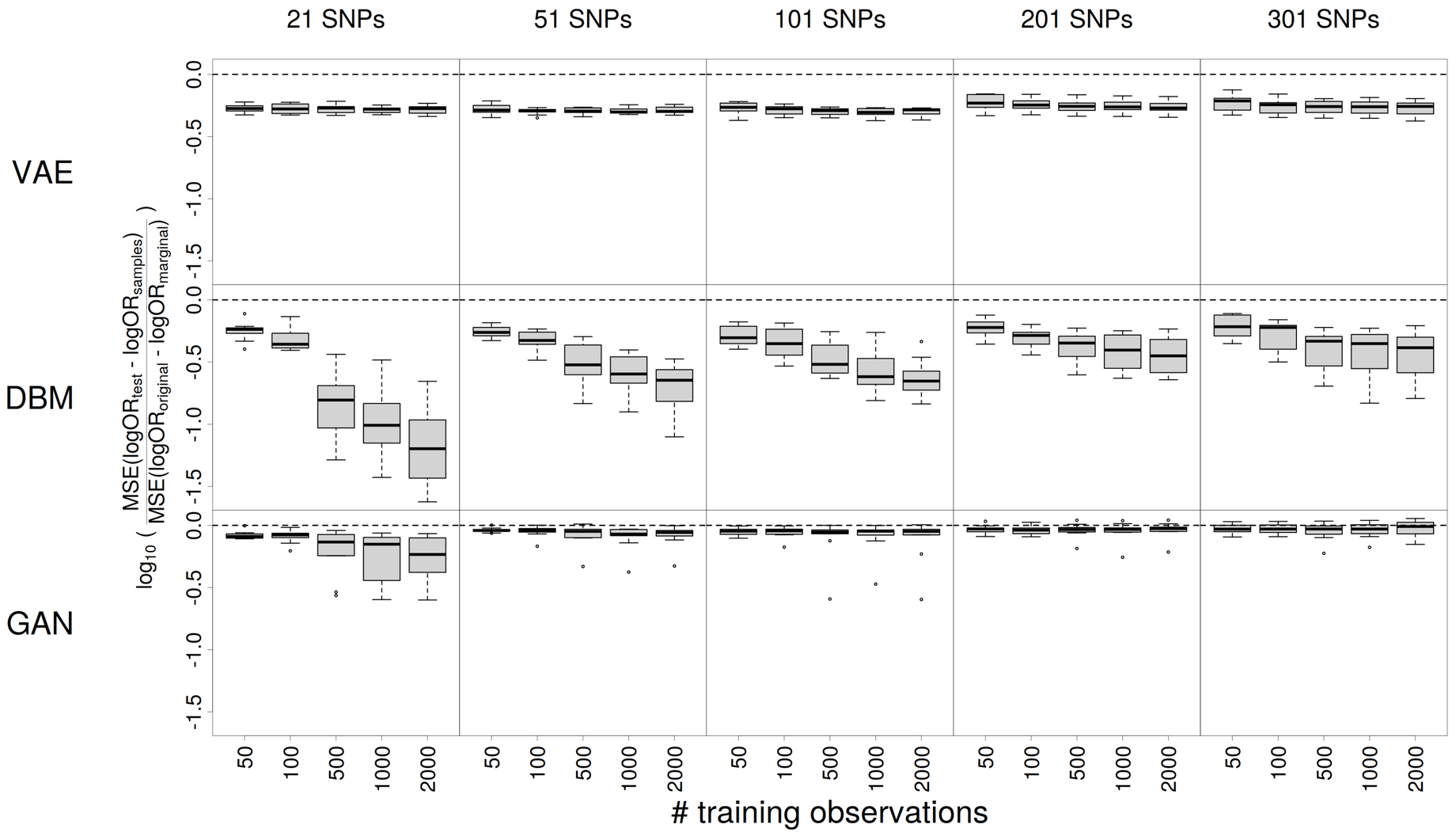
Briefings in Bioinformatics, 22(4), 2021, 1–12
https://doi.org/10.1093/bib/bbaa226
Method Review

Synthetic observations from deep generative models and binary omics data with limited sample size

Jens Nußberger†, Frederic Boesel†, Stefan Lenz, Harald Binder and Moritz Hess

small data
MORE FROM LESS

Sample size and approach performance



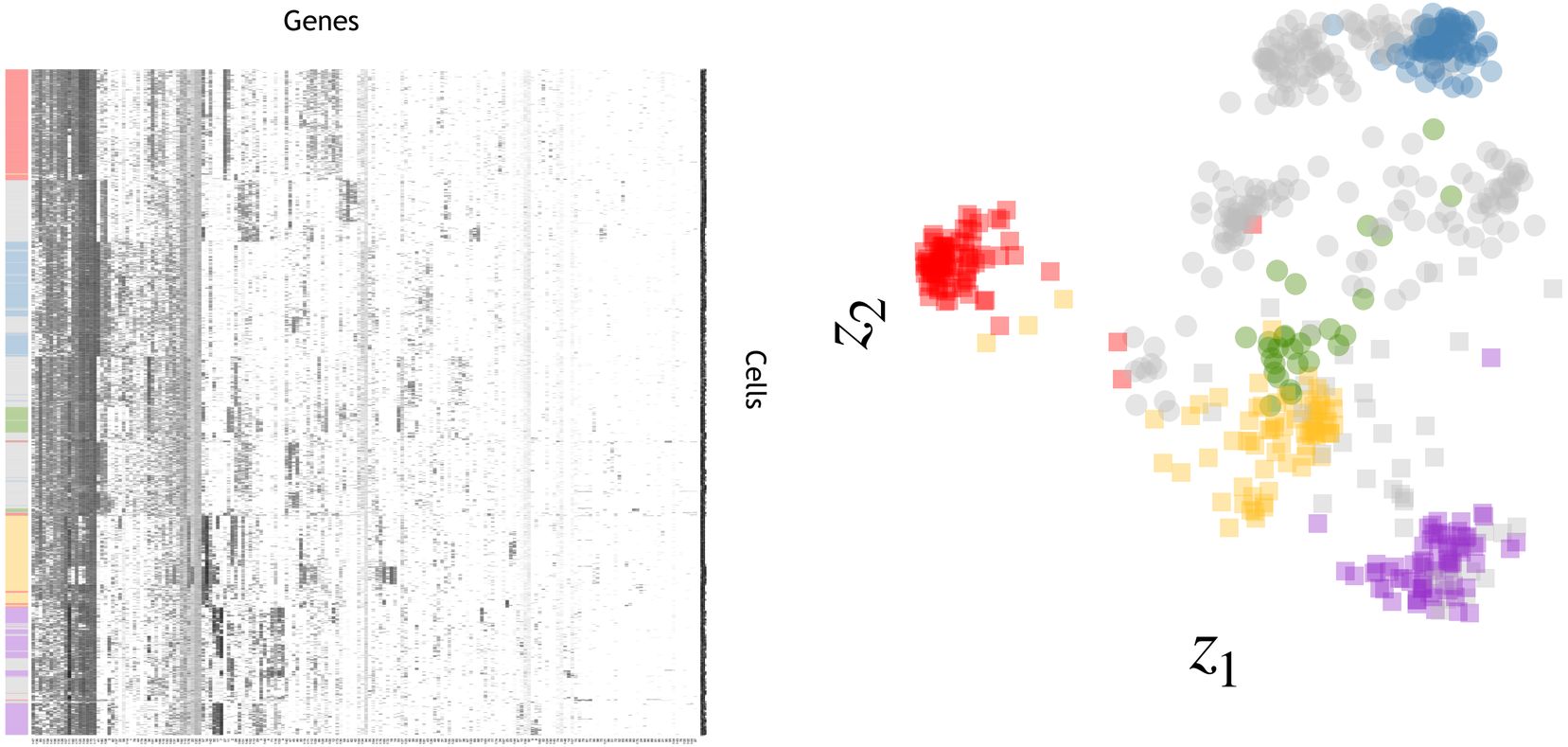
small data
MORE FROM LESS

Practical 2: setting up DGMs from scratch, model diagnostics (60 min)

- Comparing AE and VAE
  - Latent representations
  - Overfitting dependent on training data
- Judging models quality trough their generative properties
  - Accuracy of pairwise correlations
  - Similarity synthetic vs. real data

Part 3: Interpreting DGMs (Lecture 30 min, practicals 45 min)

Understanding the relationship between latent representations and the observed variables (genes).



Approaches for interpretability

- Model based
  - LDVAE
- Post-hoc
  - PatternExtractor
  - Integrated Gradients

Linearly decoded VAE (LDVAE)

model based explainability

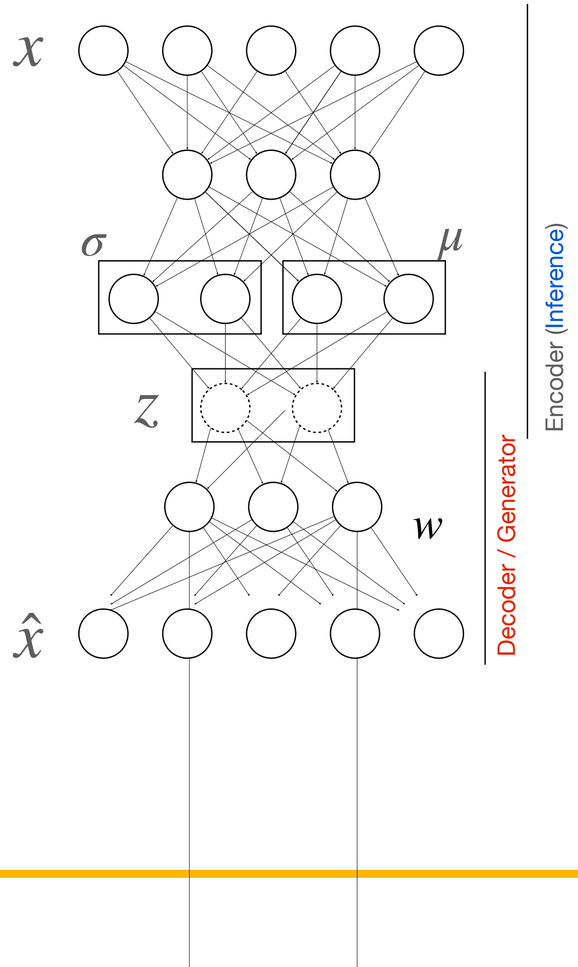
JOURNAL ARTICLE

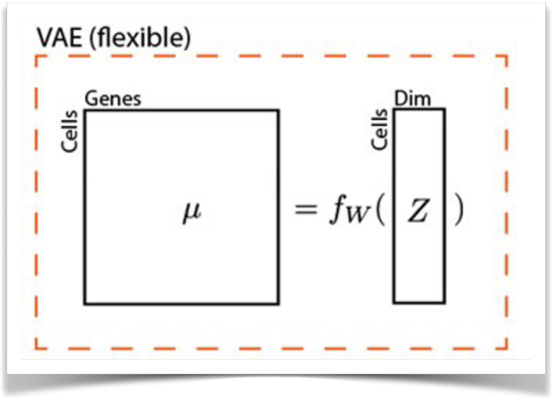
Interpretable factor models of single-cell RNA-seq via variational autoencoders

Valentine Svensson ✉, Adam Gayoso, Nir Yosef, Lior Pachter

Bioinformatics, Volume 36, Issue 11, June 2020, Pages 3418–3421,
https://doi.org/10.1093/bioinformatics/btaa169
Published: 16 March 2020 **Article history** ▾

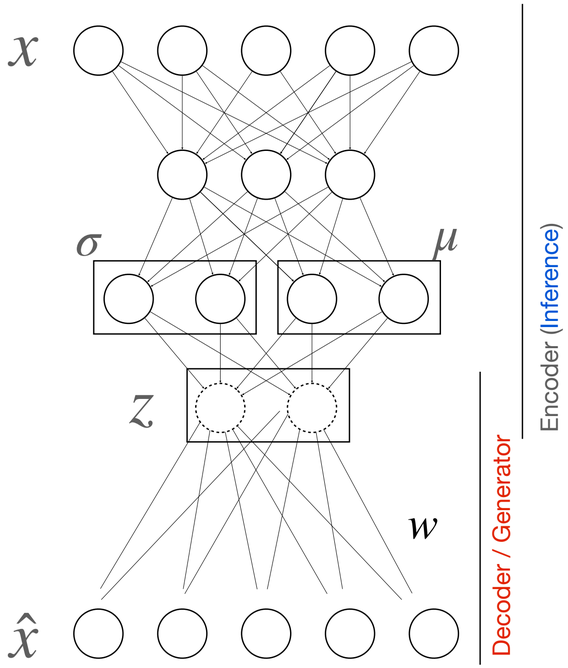
Linearly decoded VAE (LDVAE)

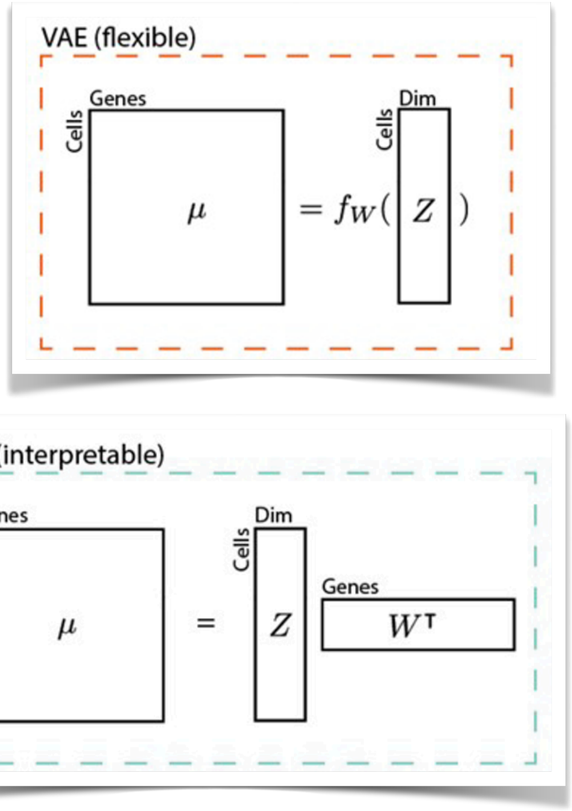




small data
MORE FROM LESS

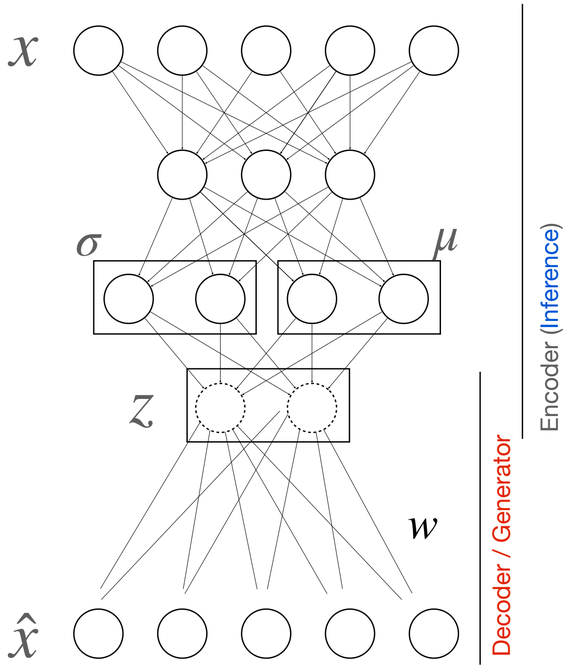
Linearly decoded VAE (LDVAE)

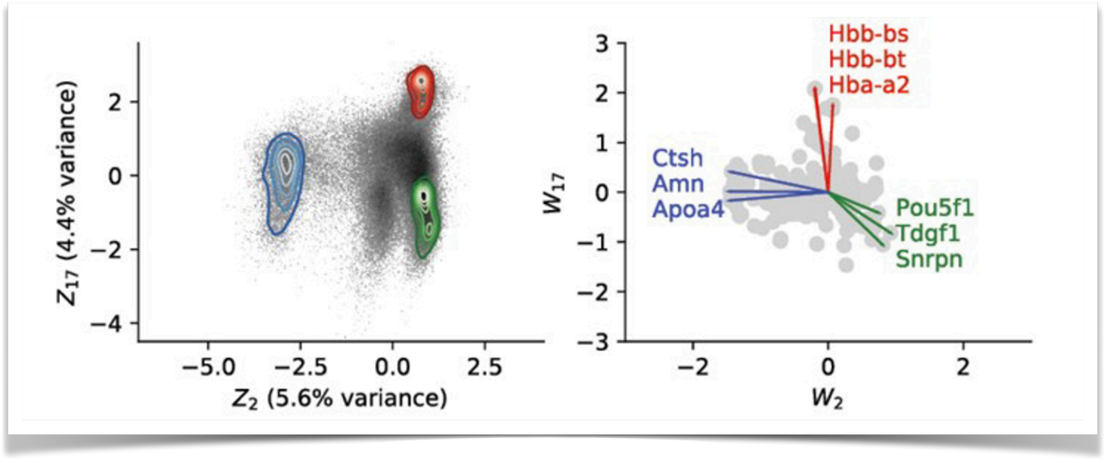




small data
MORE FROM LESS

Linearly decoded VAE (LDVAE)





small data
MORE FROM LESS

PatternExtractor: Investigating the joint distribution of discretised synthetic observations

post-hoc explainability

JOURNAL ARTICLE

Exploring generative deep learning for omics data using log-linear models

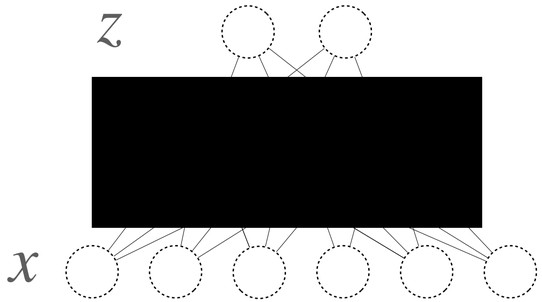
Moritz Hess ✉, Maren Hackenberg, Harald Binder

Bioinformatics, Volume 36, Issue 20, 15 October 2020, Pages 5045–5053,
https://doi.org/10.1093/bioinformatics/btaa623
Published: 01 August 2020 **Article history** ▾

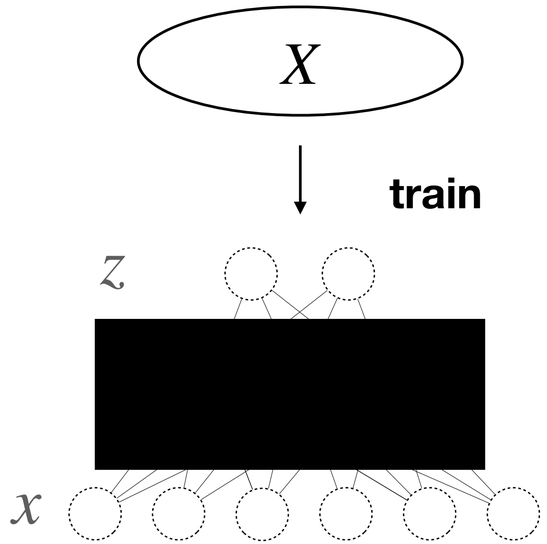
small data
MORE FROM LESS

PatternExtractor: Investigating the joint distribution of discretised synthetic observations

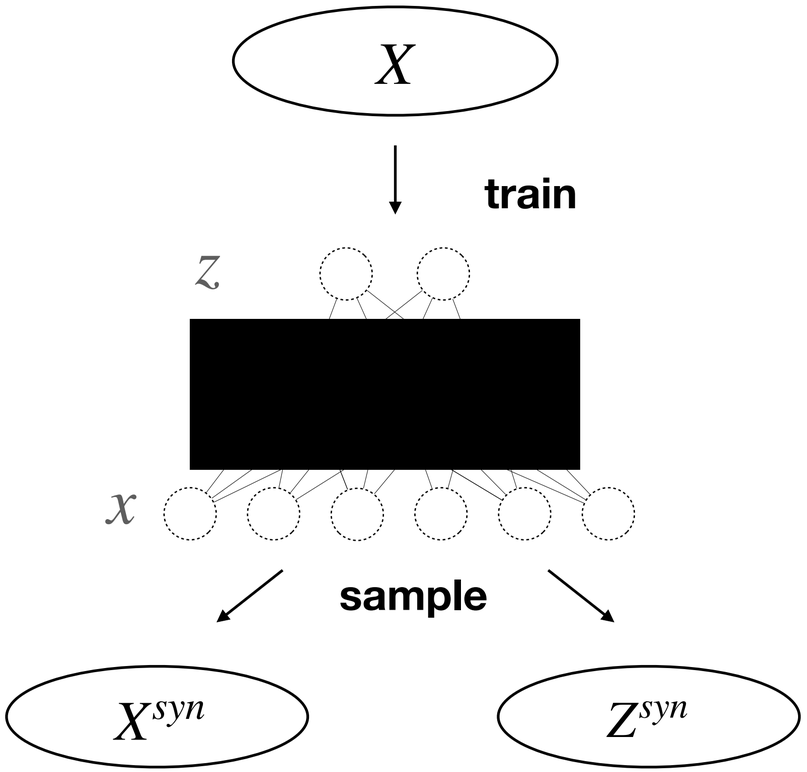
PatternExtractor: Investigating the joint distribution of discretised synthetic observations



PatternExtractor: Investigating the joint distribution of discretised synthetic observations

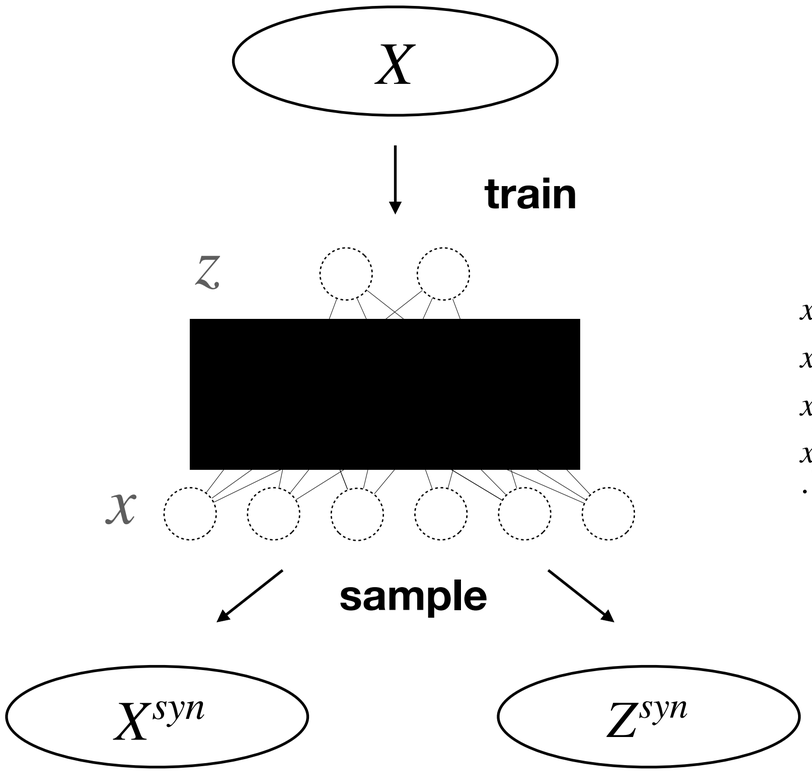


PatternExtractor: Investigating the joint distribution of discretised synthetic observations



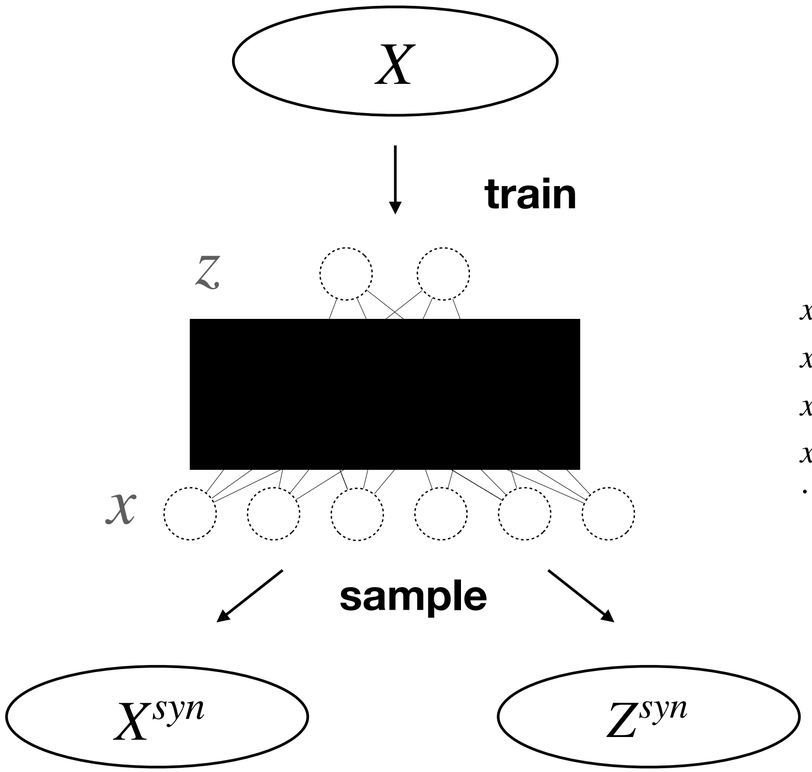
small data
MORE FROM LESS

PatternExtractor: Investigating the joint distribution of discretised synthetic observations



$$x^{syn}_{1.} = 12\ 32\ 4\ 5\ 13\ 5 \qquad z^{syn}_{1.} = -3.2\ 2.8$$
$$x^{syn}_{2.} = 68\ 90\ 8\ 44\ 31\ 9 \qquad z^{syn}_{2.} = -1.2\ 1.4$$
$$x^{syn}_{3.} = 6\ 5\ 78\ 12\ 24\ 36 \qquad z^{syn}_{3.} = 1.2\ -1.2$$
$$x^{syn}_{4.} = 98\ 5\ 6\ 76\ 47\ 67 \qquad z^{syn}_{4.} = 4.2\ -2.0$$
$$\cdots \qquad \cdots$$

PatternExtractor: Investigating the joint distribution of discretised synthetic observations



$$x^{syn}_{1.} = 0\ 1\ 0\ 0\ 1\ 0 \qquad z^{syn}_{1.} = 0\ 1$$
$$x^{syn}_{2.} = 1\ 1\ 0\ 1\ 1\ 0 \qquad z^{syn}_{2.} = 0\ 1$$
$$x^{syn}_{3.} = 0\ 0\ 1\ 0\ 1\ 1 \qquad z^{syn}_{3.} = 1\ 0$$
$$x^{syn}_{4.} = 1\ 0\ 0\ 1\ 1\ 1 \qquad z^{syn}_{4.} = 1\ 0$$
$$\cdots \qquad \cdots$$

Investigating the joint distribution of discretized synthetic observations

Investigating the joint distribution of discretized synthetic observations

$$
\begin{aligned}
x^{syn}_{1.} &= 0\ 1\ 0\ 0\ 1\ 0 \qquad z^{syn}_{1.} = 0\ 1\\
x^{syn}_{2.} &= 1\ 1\ 0\ 1\ 1\ 0 \qquad z^{syn}_{2.} = 0\ 1\\
x^{syn}_{3.} &= 0\ 0\ 1\ 0\ 1\ 1 \qquad z^{syn}_{3.} = 1\ 0\\
x^{syn}_{4.} &= 1\ 0\ 0\ 1\ 1\ 1 \qquad z^{syn}_{4.} = 1\ 0\\
&\cdots \qquad\qquad \cdots
\end{aligned}
$$

Investigating the joint distribution of discretized synthetic observations

$$
\begin{array}{llllllll}
x_{1.}^{syn} = & 0 & 1 & 0 & 0 & 1 & 0 & \\
x_{2.}^{syn} = & 1 & 1 & 0 & 1 & 1 & 0 & \\
x_{3.}^{syn} = & 0 & 0 & 1 & 0 & 1 & 1 & \\
x_{4.}^{syn} = & 1 & 0 & 0 & 1 & 1 & 1 & \\
\cdots &&&&&&&
\end{array}
\qquad
\begin{array}{lll}
z_{1.}^{syn} = & 0 & 1 \\
z_{2.}^{syn} = & 0 & 1 \\
z_{3.}^{syn} = & 1 & 0 \\
z_{4.}^{syn} = & 1 & 0 \\
\cdots &&
\end{array}
$$

	$X_*^{syn} = 0$	$X_*^{syn} = 1$
$Z_1^{syn} = 0$		
$Z_1^{syn} = 1$		

Investigating the joint distribution of discretized synthetic observations

$$
\begin{array}{llllllll}
x^{syn}_{1.} = & 0 & 1 & 0 & 0 & 1 & 0 & \qquad z^{syn}_{1.} = 0 \quad 1 \\
x^{syn}_{2.} = & 1 & 1 & 0 & 1 & 1 & 0 & \qquad z^{syn}_{2.} = 0 \quad 1 \\
x^{syn}_{3.} = & 0 & 0 & 1 & 0 & 1 & 1 & \qquad z^{syn}_{3.} = 1 \quad 0 \\
x^{syn}_{4.} = & 1 & 0 & 0 & 1 & 1 & 1 & \qquad z^{syn}_{4.} = 1 \quad 0 \\
\cdots & & & & & & & \qquad \cdots
\end{array}
$$

	$X^{syn}_{*} = 0$	$X^{syn}_{*} = 1$
$Z^{syn}_{1} = 0$		
$Z^{syn}_{1} = 1$		

$$log(\mu_{i,j,k}) = \lambda + \lambda^{X_2}_{i} + \lambda^{X_6}_{j} + \lambda^{U}_{k} + \lambda^{X_2X_6}_{ij} + \lambda^{X_2U}_{ik} + \lambda^{X_6U}_{jk}$$

$$i, j, k \in \{1,2\}$$

Investigating the joint distribution of discretized synthetic observations

$$
\begin{array}{llllllll}
x^{syn}_{1.} = & 0 & 1 & 0 & 0 & 1 & 0 & \\
x^{syn}_{2.} = & 1 & 1 & 0 & 1 & 1 & 0 & \\
x^{syn}_{3.} = & 0 & 0 & 1 & 0 & 1 & 1 & \\
x^{syn}_{4.} = & 1 & 0 & 0 & 1 & 1 & 1 & \\
\cdots &&&&&&&
\end{array}
\qquad
\begin{array}{lll}
z^{syn}_{1.} = & 0 & 1 \\
z^{syn}_{2.} = & 0 & 1 \\
z^{syn}_{3.} = & 1 & 0 \\
z^{syn}_{4.} = & 1 & 0 \\
\cdots &&
\end{array}
$$

	$X^{syn}_{*} = 0$	$X^{syn}_{*} = 1$
$Z^{syn}_{1} = 0$		
$Z^{syn}_{1} = 1$		

$$log(\mu_{i,j,k}) = \lambda + \lambda^{X_2}_{i} + \lambda^{X_6}_{j} + \lambda^{U}_{k} + \lambda^{X_2X_6}_{ij} + \lambda^{X_2U}_{ik} + \lambda^{X_6U}_{jk}$$

$$i, j, k \in \{1,2\}$$

Investigating the joint distribution of discretized synthetic observations

$$
\begin{array}{llllllll}
x^{syn}_{1.} = & 0 & 1 & 0 & 0 & 1 & 0 & \qquad z^{syn}_{1.} = 0 \quad 1 \\
x^{syn}_{2.} = & 1 & 1 & 0 & 1 & 1 & 0 & \qquad z^{syn}_{2.} = 0 \quad 1 \\
x^{syn}_{3.} = & 0 & 0 & 1 & 0 & 1 & 1 & \qquad z^{syn}_{3.} = 1 \quad 0 \\
x^{syn}_{4.} = & 1 & 0 & 0 & 1 & 1 & 1 & \qquad z^{syn}_{4.} = 1 \quad 0 \\
\cdots & & & & & & & \qquad \cdots
\end{array}
$$

	$X^{syn}_{*} = 0$	$X^{syn}_{*} = 1$
$Z^{syn}_{1} = 0$		
$Z^{syn}_{1} = 1$		

$$log(\mu_{i,j,k}) = \lambda + \lambda^{X_2}_{i} + \lambda^{X_6}_{j} + \lambda^{U}_{k} + \lambda^{X_2X_6}_{ij} + \lambda^{X_2U}_{ik} \,.$$

$$i, j, k \in \{1,2\}$$

Investigating the joint distribution of discretized synthetic observations

$$
\begin{array}{llllllll}
x^{syn}_{1.} = & 0 & 1 & 0 & 0 & 1 & 0 & \qquad z^{syn}_{1.} = 0 \quad 1 \\
x^{syn}_{2.} = & 1 & 1 & 0 & 1 & 1 & 0 & \qquad z^{syn}_{2.} = 0 \quad 1 \\
x^{syn}_{3.} = & 0 & 0 & 1 & 0 & 1 & 1 & \qquad z^{syn}_{3.} = 1 \quad 0 \\
x^{syn}_{4.} = & 1 & 0 & 0 & 1 & 1 & 1 & \qquad z^{syn}_{4.} = 1 \quad 0 \\
\cdots & & & & & & & \qquad \cdots
\end{array}
$$

	$X^{syn}_{*} = 0$	$X^{syn}_{*} = 1$
$Z^{syn}_{1} = 0$		
$Z^{syn}_{1} = 1$		

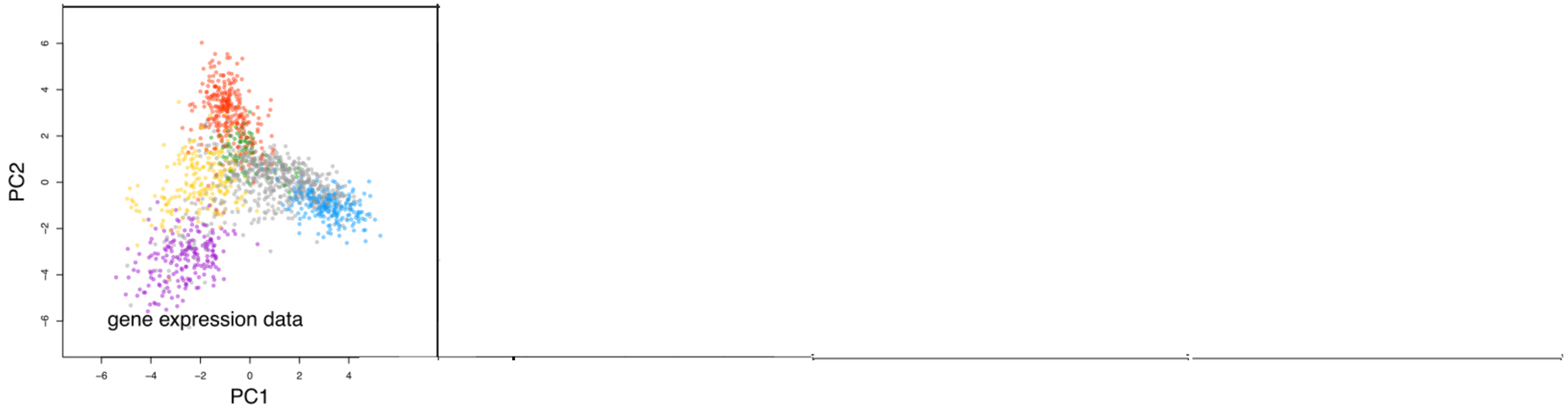
$$log(\mu_{i,j,k}) = \lambda + \lambda_i^{X_2} + \lambda_j^{X_6} + \lambda_k^{U} + \lambda_{ij}^{X_2X_6} + \lambda_{ik}^{X_2U} \cdot$$

$$log(\mu_{i,j,k}) = \lambda + \lambda_i^{X_2} + \lambda_j^{X_6} + \lambda_k^{U} + \lambda_{ij}^{X_2X_6} + \lambda_{ik}^{X_2U} + \lambda_{jk}^{X_6U}$$

$$i, j, k \in \{1,2\}$$

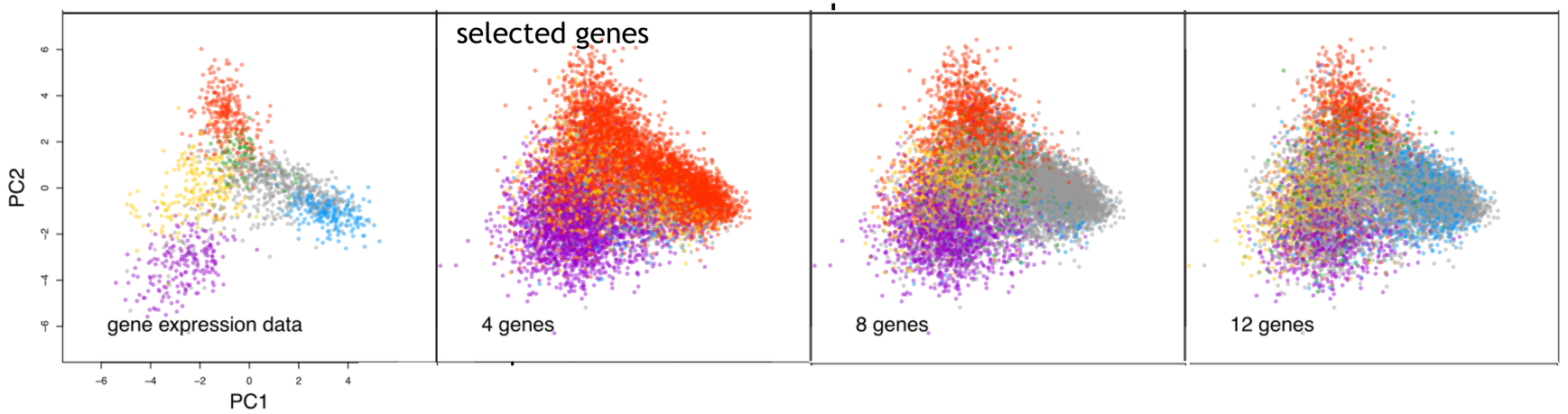
Patterns in selected genes allow to assign synthetic data to their overall most similar training example

Synthetic data matched to training data based on patterns in ...



Patterns in selected genes allow to assign synthetic data to their overall most similar training example

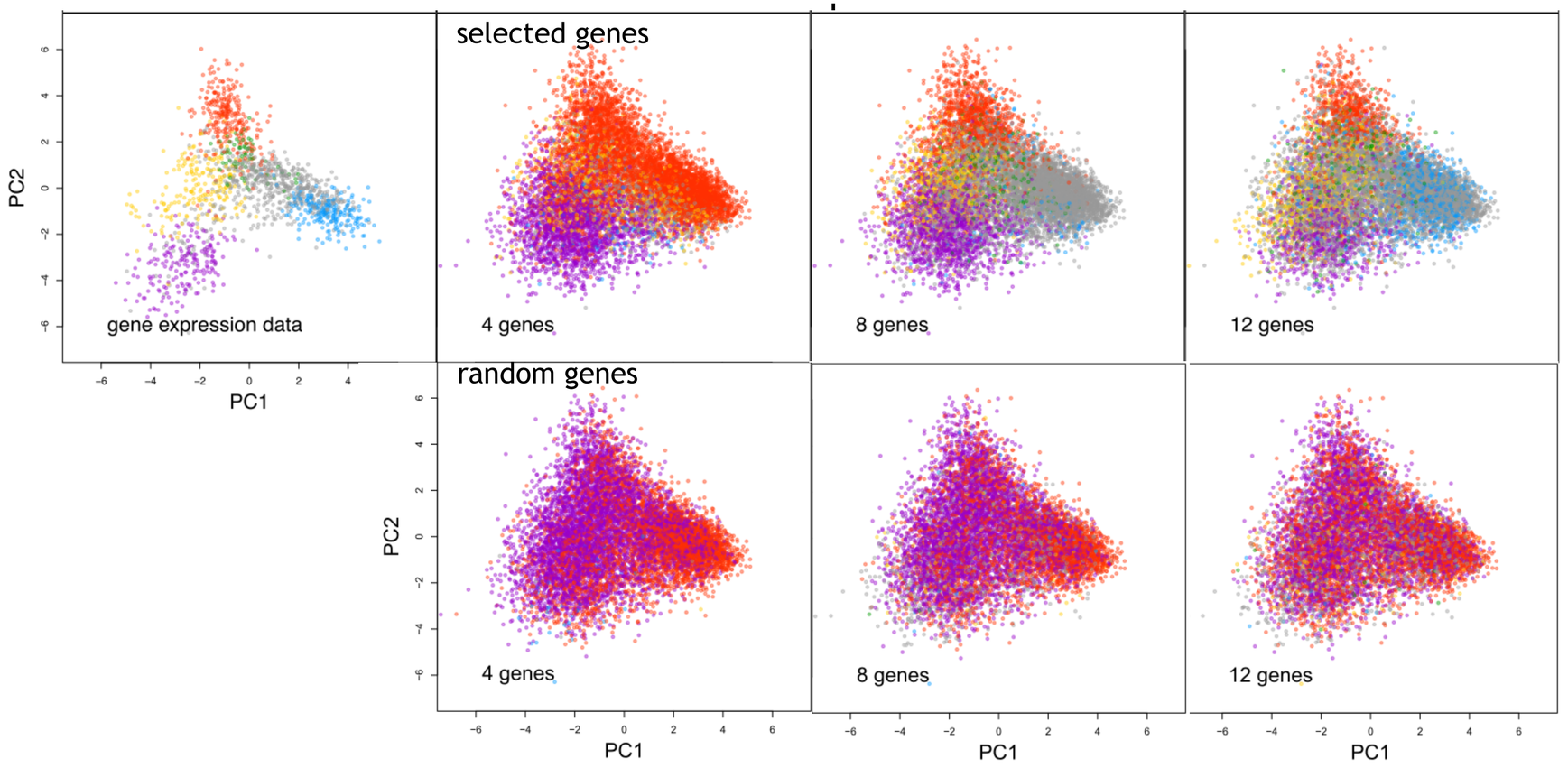
Synthetic data matched to training data based on patterns in ...



small data
MORE FROM LESS

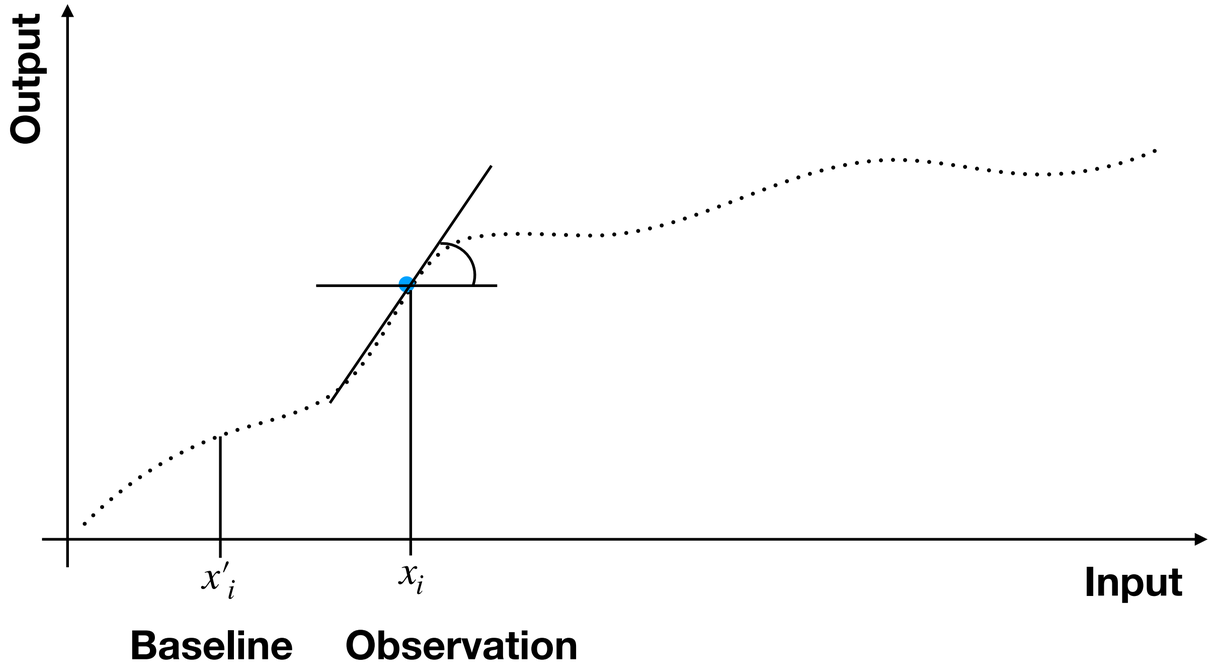
Patterns in selected genes allow to assign synthetic data to their overall most similar training example

Synthetic data matched to training data based on patterns in ...



small data
MORE FROM LESS

Integrated Gradients



Axiomatic Attribution for Deep Networks

Mukund Sundararajan [*1] **Ankur Taly** [*1] **Qiqi Yan** [*1]

interpolate m observations at k intervals

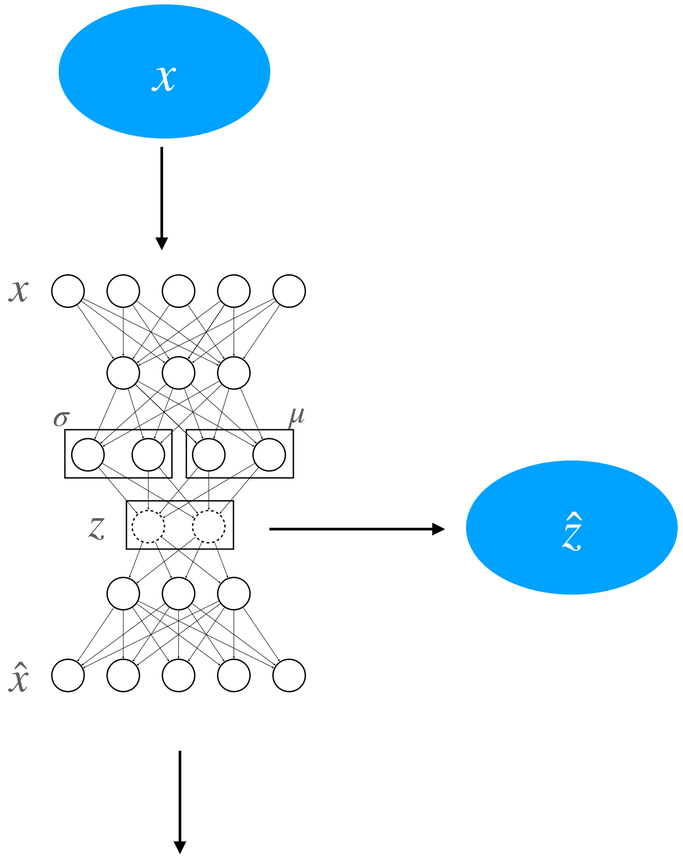
$$IntegratedGrads_i^{approx}(x) ::= (x_i - x'_i) \times \sum_{k=1}^{m} \frac{\partial F(\overbrace{x' + \frac{k}{m} \times (x - x')}^{\text{interpolate m observations at k intervals}})}{\partial x_i} \times \frac{1}{m}$$

Practical 3: interpreting DGMs (45 min)

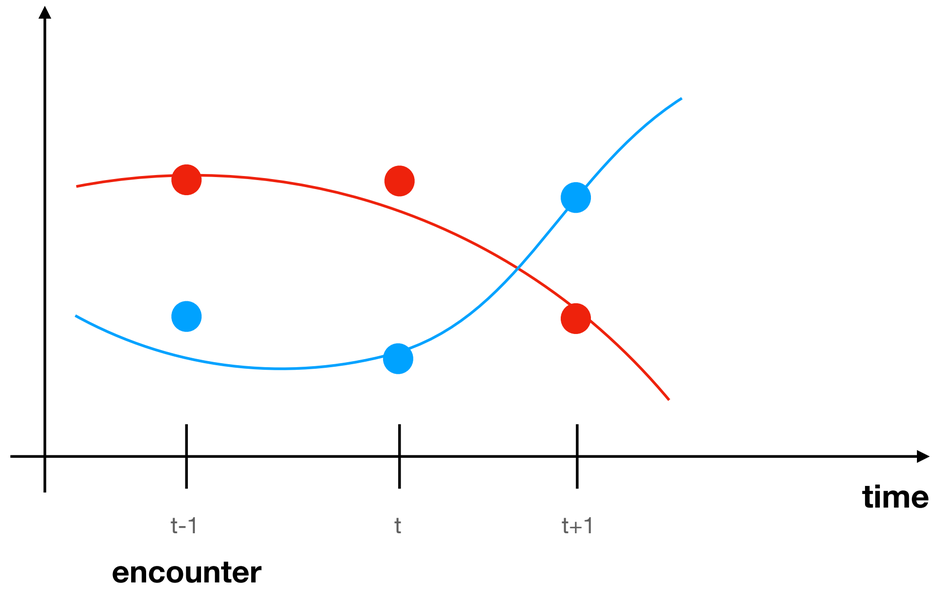
- Modify a VAE into an LDVAE
- Compare LDVAE and VAE
- Identify influential genes with LDVAE
- Identify influential genes with integrated gradients (IG)
- Compare LDVAE with IG results

Part 4: Recapitulation, Outlook (Lecture 15 min)

What can you do with deep generative models



- Similarity between observations
- Combining with classical model, e.g. dynamic model



- augment data
- share data
- smooth data

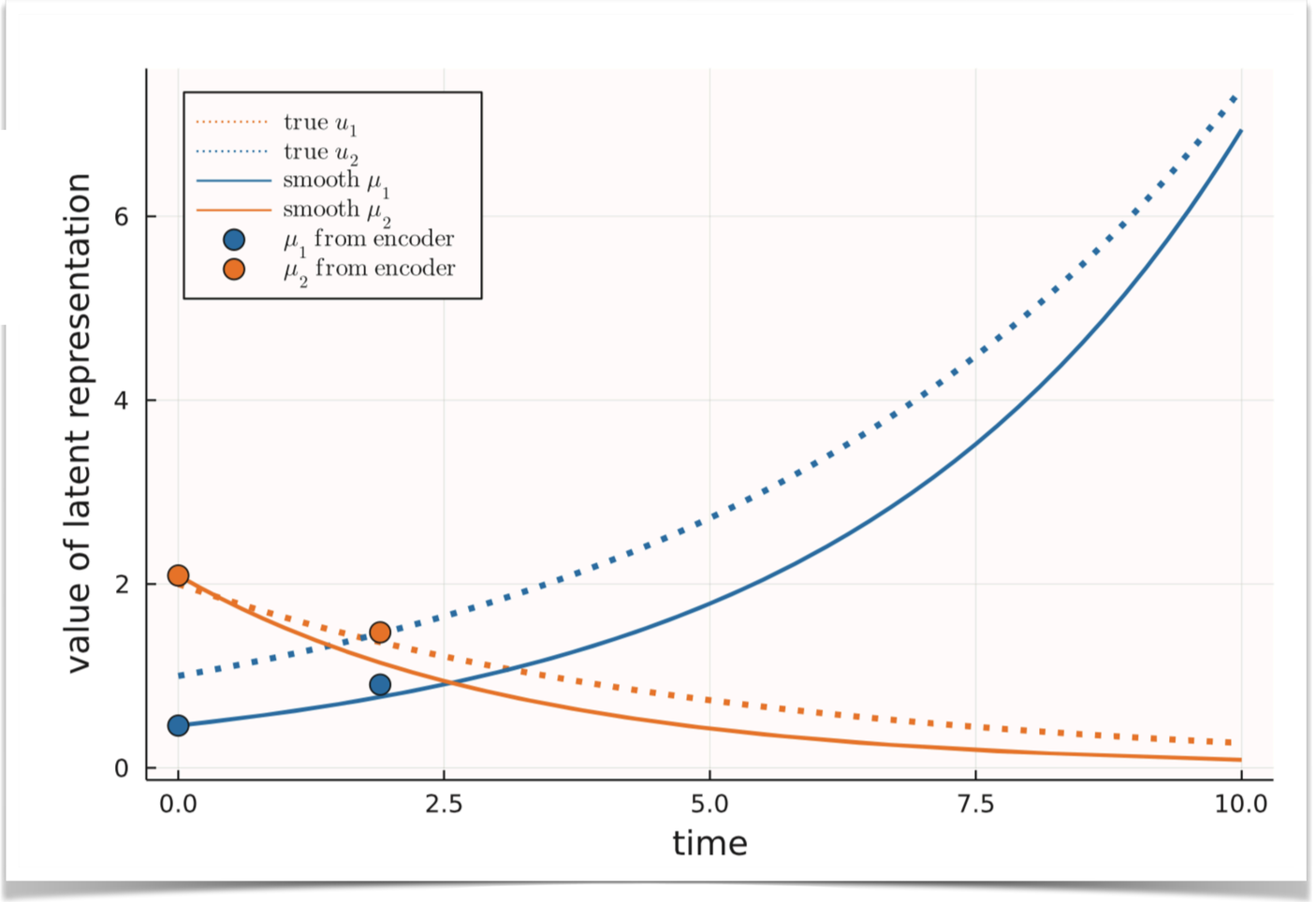
- Missing data:
  - Synthetic control group
  - Imputing missing values

small data
MORE FROM LESS

Longitudinal clinical data

Deep dynamic modeling with just two time points: Can we still allow for individual trajectories?

Maren Hackenberg[1] | Philipp Harms[2] | Michelle Pfaffenlehner[1] | Astrid Pechmann[3] | Janbernd Kirschner[3,4] | Thorsten Schmidt[2] | Harald Binder[1]



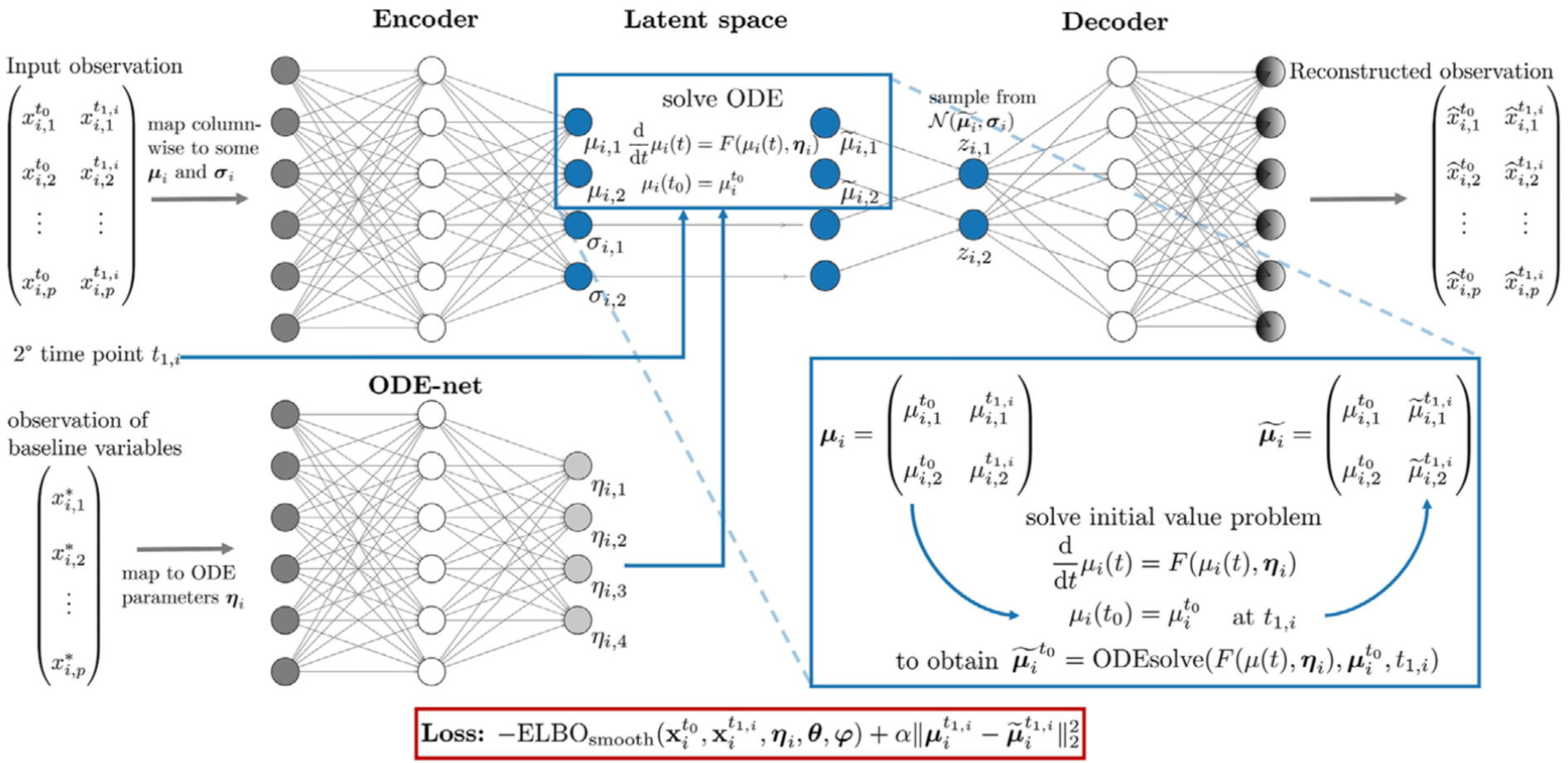
small data
MORE FROM LESS

Longitudinal clinical data

Deep dynamic modeling with just two time points: Can we still allow for individual trajectories?

Maren Hackenberg[1] | Philipp Harms[2] | Michelle Pfaffenlehner[1] | Astrid Pechmann[3] | Janbernd Kirschner[3,4] | Thorsten Schmidt[2] | Harald Binder[1]

Longitudinal clinical data



small data
MORE FROM LESS


PORT
SHOP
Get more, pay less.
extra
END OF
Part 1