



# Deep Learning for Healthcare

**Autoencoder**

*Jimeng Sun*

# Outline



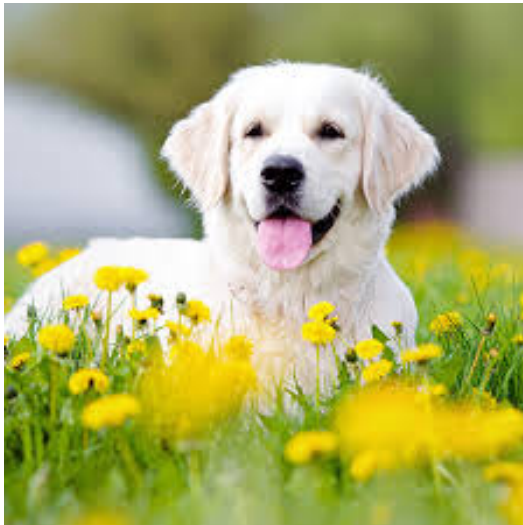
- Autoencoder
  - Variants of autoencoders
    - Sparse autoencoder
    - Denoising autoencoder
    - Stacked autoencoder
  - Healthcare applications of autoencoders

# Autoencoders



- Compression & decompression
  - Learning the latent representation of a given sample  $x$

256 X 256 dimensions



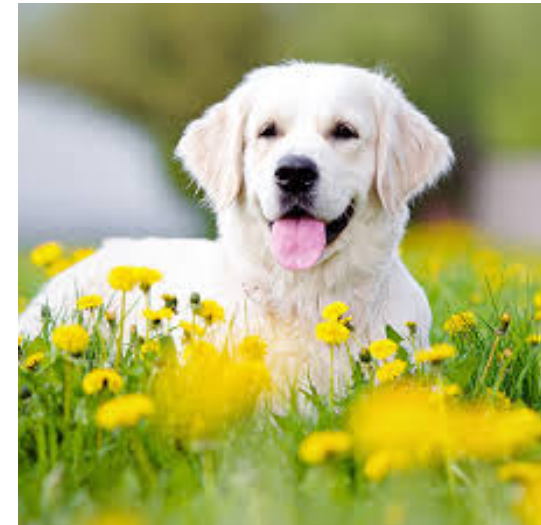
→  
encoder

128 dimensions

0.1  
0.2  
-0.4  
1.5  
-2.1  
0.2  
...  
...

→  
decoder

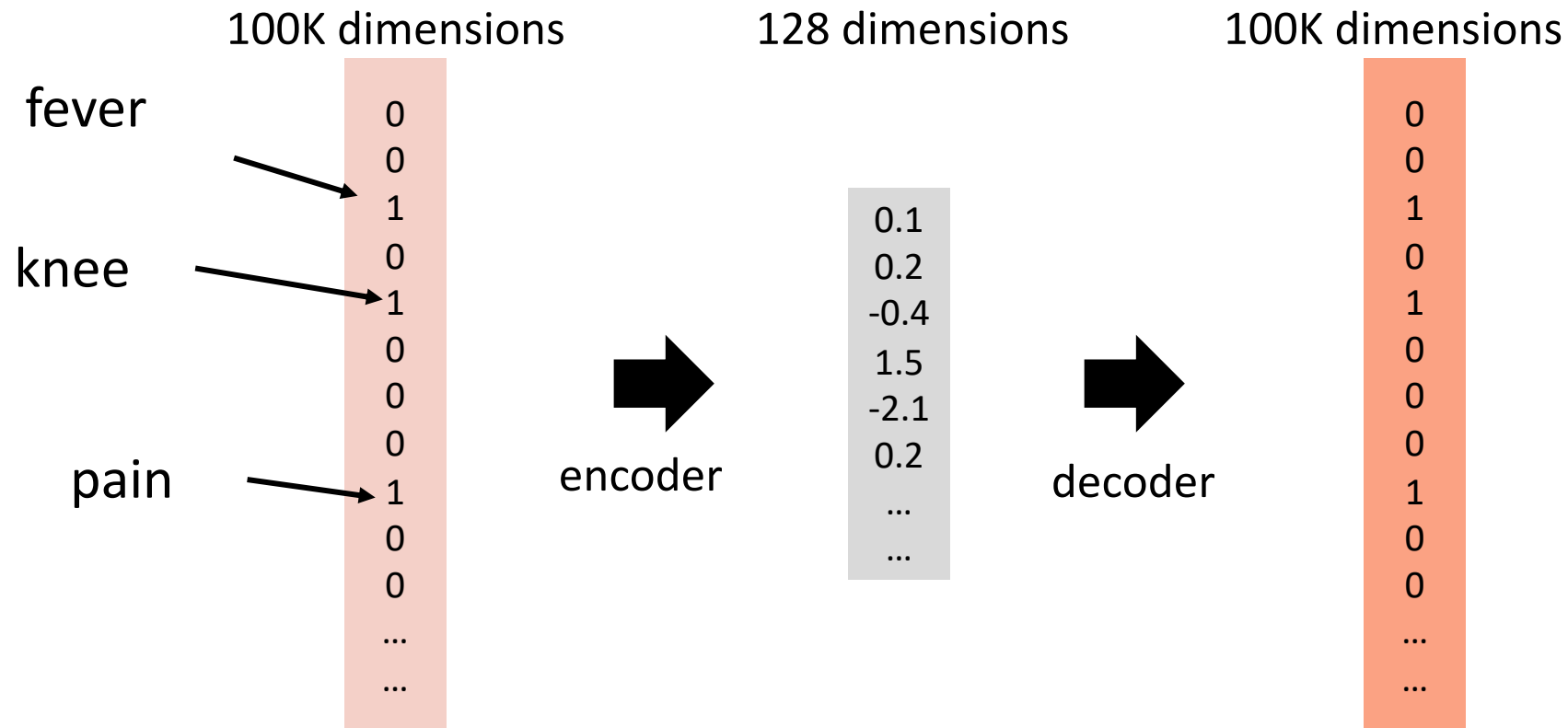
256 X 256 dimensions



# Autoencoders



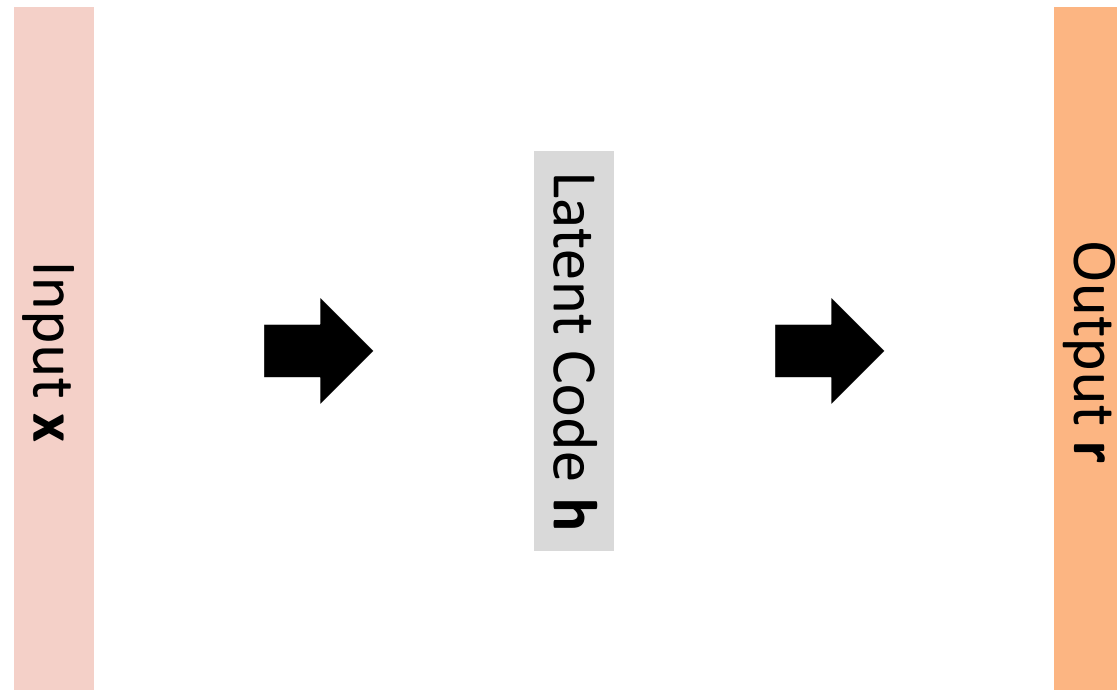
- Learning the latent representation of a given sample  $\mathbf{x}$



# Autoencoders



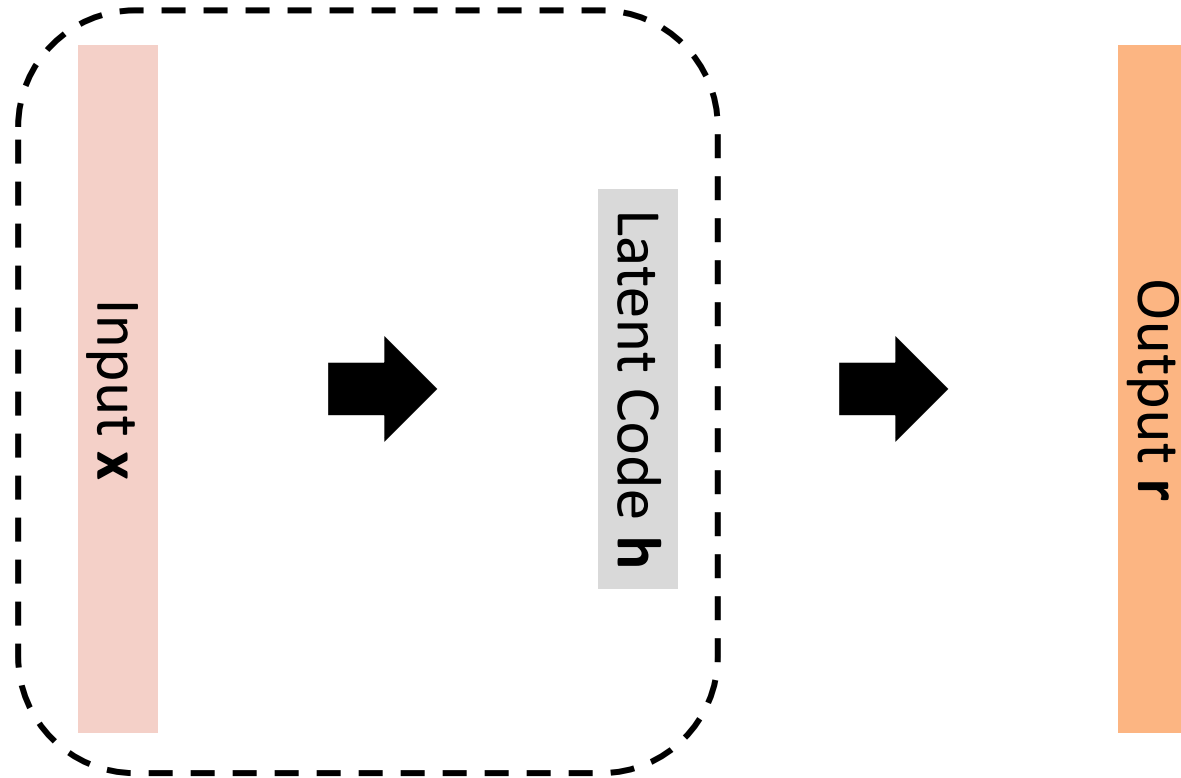
- Learning the latent representation of a given sample  $\mathbf{x}$



# Autoencoders



- Learning the latent representation of a given sample  $\mathbf{x}$

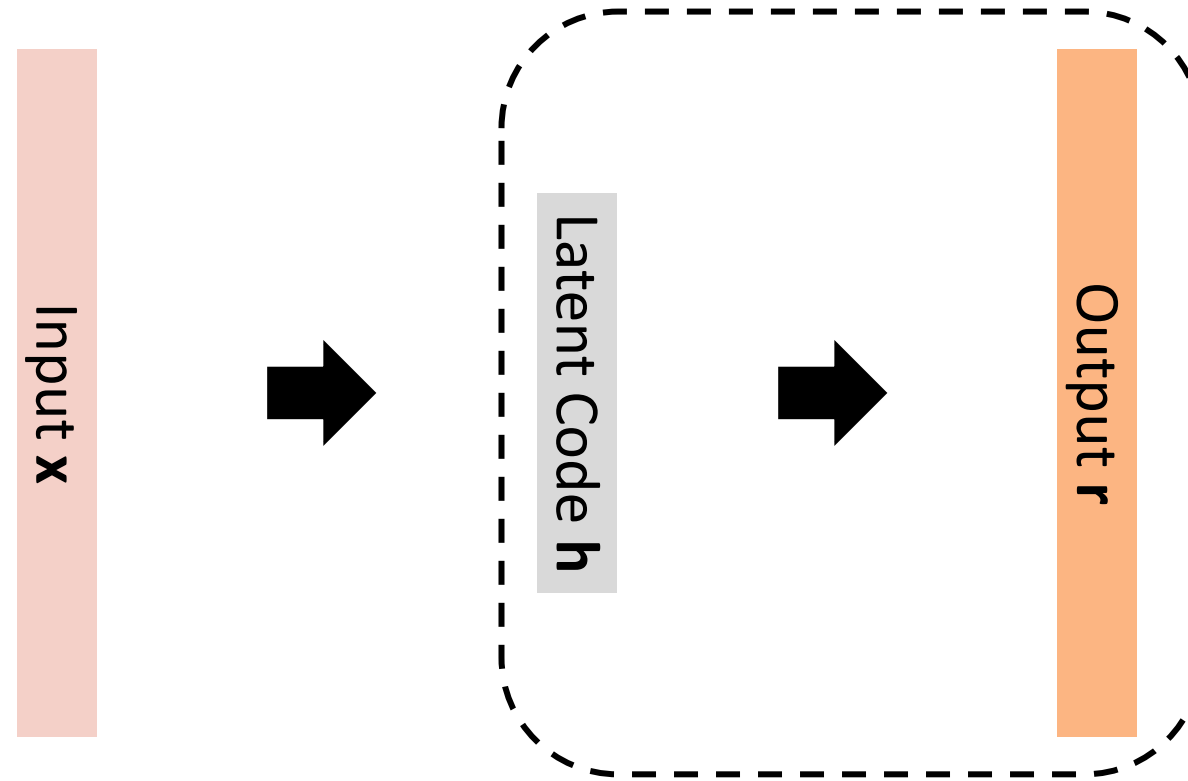


Encoding  $\mathbf{h} = f_{\theta}(\mathbf{x}) = \sigma_1(\mathbf{W}\mathbf{x} + \mathbf{b})$

# Autoencoders

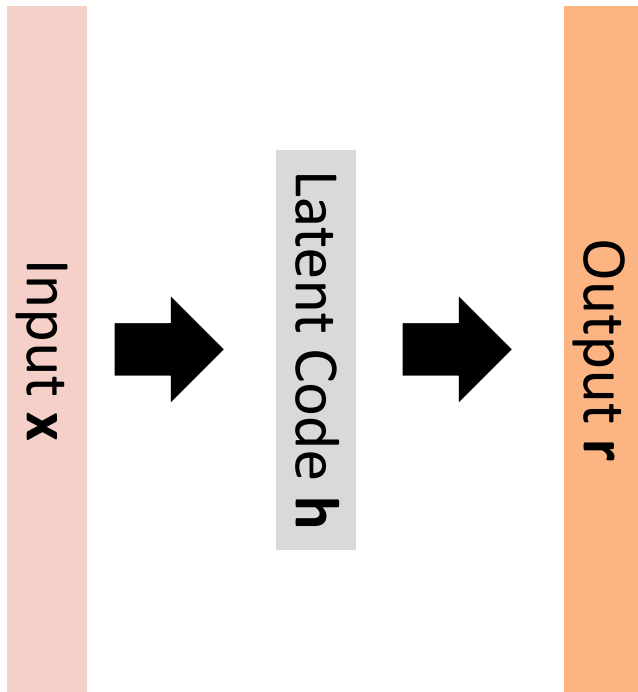


- Learning the latent representation of a given sample  $\mathbf{x}$



Decoding  $\mathbf{r} = g_{\theta'}(\mathbf{h}) = \sigma_2(\mathbf{W}'\mathbf{h} + \mathbf{b}')$

# Autoencoders



Minimize reconstruction error

$$\left\{ \begin{array}{l} L(\mathbf{x}, \mathbf{r}) = \|\mathbf{x} - \mathbf{r}\|^2 \quad \text{for Gaussian input} \\ L(\mathbf{x}, \mathbf{r}) = -\sum_i [\mathbf{x}_i \log \mathbf{r}_i + (1 - \mathbf{x}_i) \log(1 - \mathbf{r}_i)] \\ \quad \text{for binary input} \end{array} \right.$$

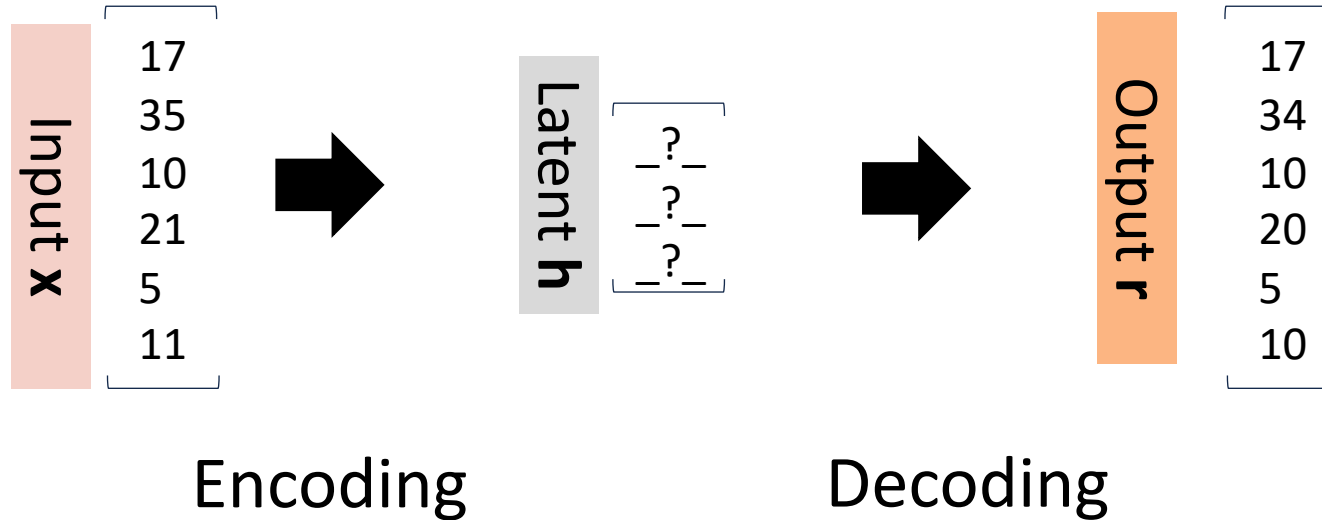
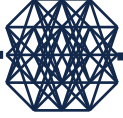


# Properties of Autoencoder



- **Unsupervised:** no labels are required
- **Data specific:** compress similar data to the training data
- **Lossy:** reconstruction will not be identical to the input

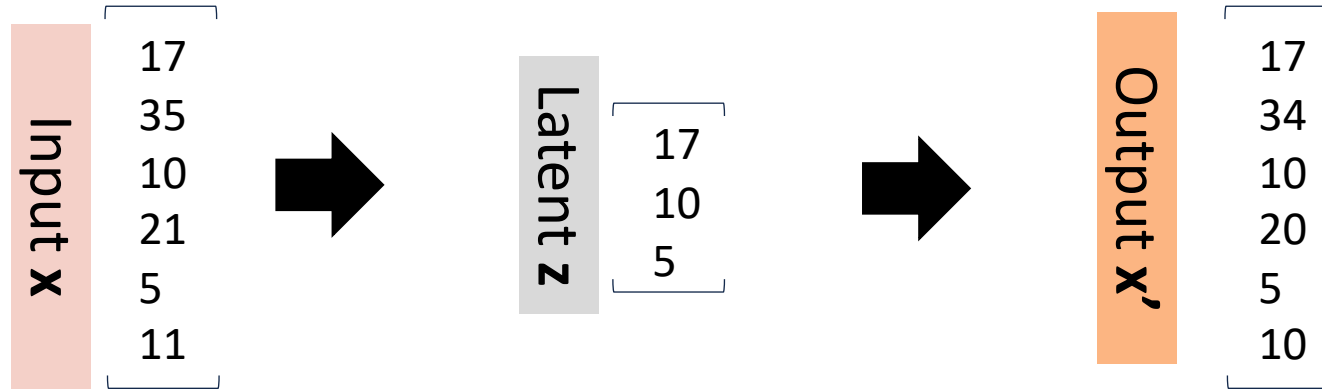
# Quiz: Autoencoder



- Compress and decompress by finding patterns
- What is the latent pattern  $h$ ?

$$h = [17, 10, 5]$$

# Quiz: Autoencoder



$$\mathbf{z} = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b})$$

Encoding

$$\mathbf{x}' = \sigma'(\mathbf{W}'\mathbf{z} + \mathbf{b}')$$

Decoding

- Assume linear activation  $\sigma$  and  $\sigma'$ 
  - What is the encoding matrix  $\mathbf{W}$ ?
  - What is the encoding matrix  $\mathbf{W}'$ ?

$$\mathbf{W} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\mathbf{W}' = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 2 \end{bmatrix}$$

# Variants of autoencoder

---

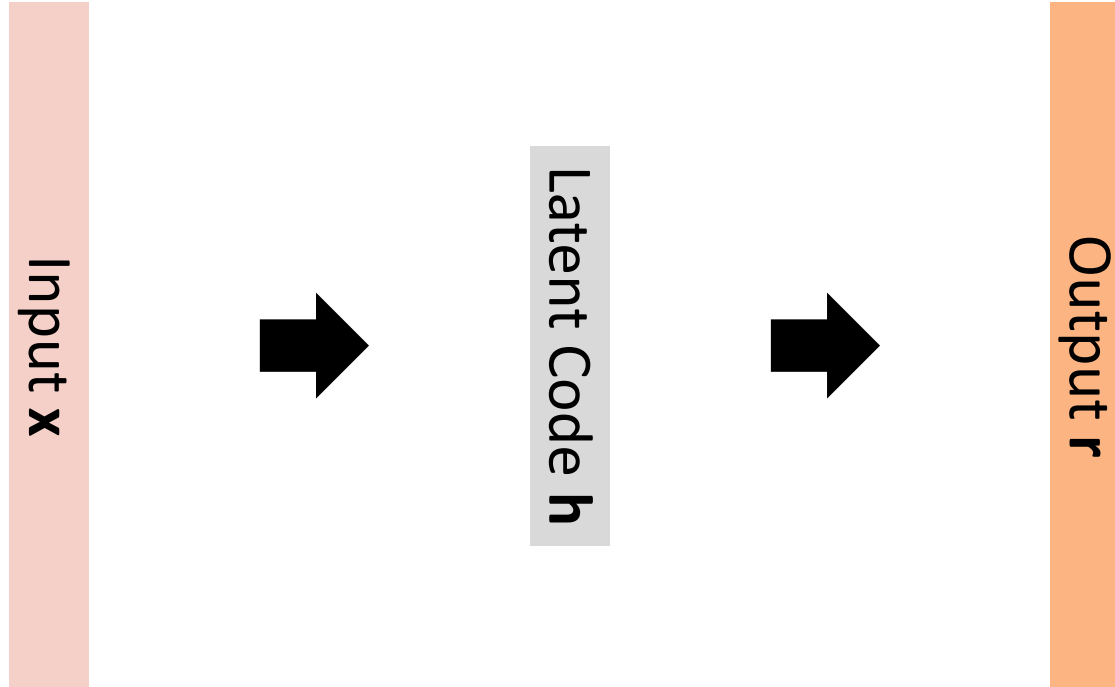


- Sparse autoencoder
- Denoising autoencoder
- Stacked autoencoder

# Sparse Autoencoder



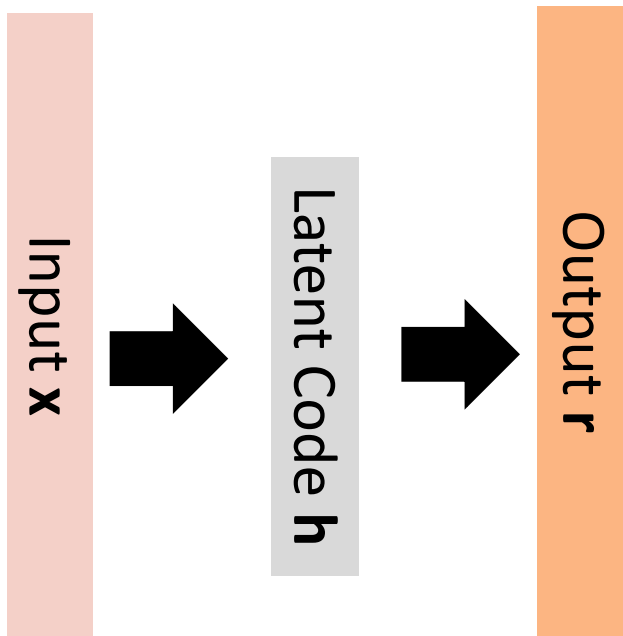
- Sparsity in  $h$



# Sparse Autoencoder: Ideas



- Sparsity in  $h$



Sigmoid activation in the hidden layer  $\sim$  between 0 and 1

$$\hat{\rho}_j = \frac{1}{n} \sum_{i=1}^n h_j[i] \quad \text{Sparsity level} = \text{Average activation}$$

Target sparsity level  $\rho = 0.05$

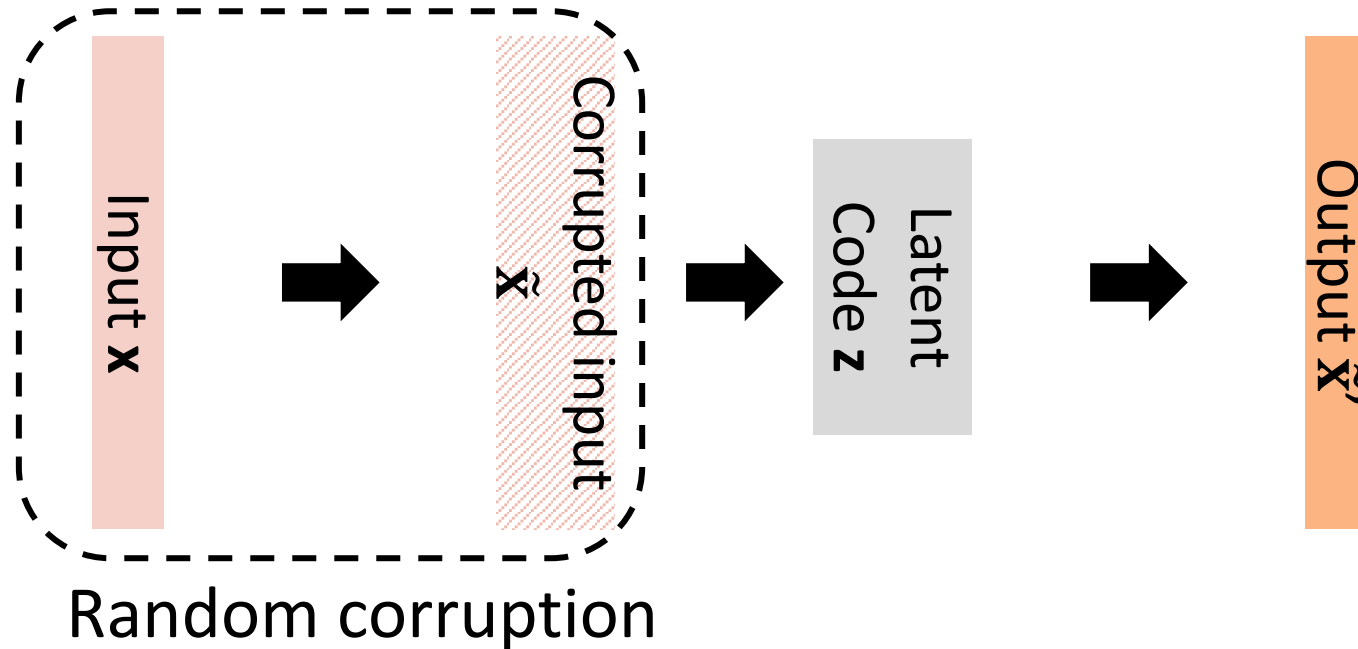
$$\arg \min_{\theta, \theta'} \frac{1}{n} \sum_{i=1}^n L(\mathbf{x}_i, \mathbf{r}_i) + \gamma \sum_{j=1}^k \mathcal{D}_{KL}(\rho || \hat{\rho}_j)$$

Regularization term

# Denoising Autoencoders



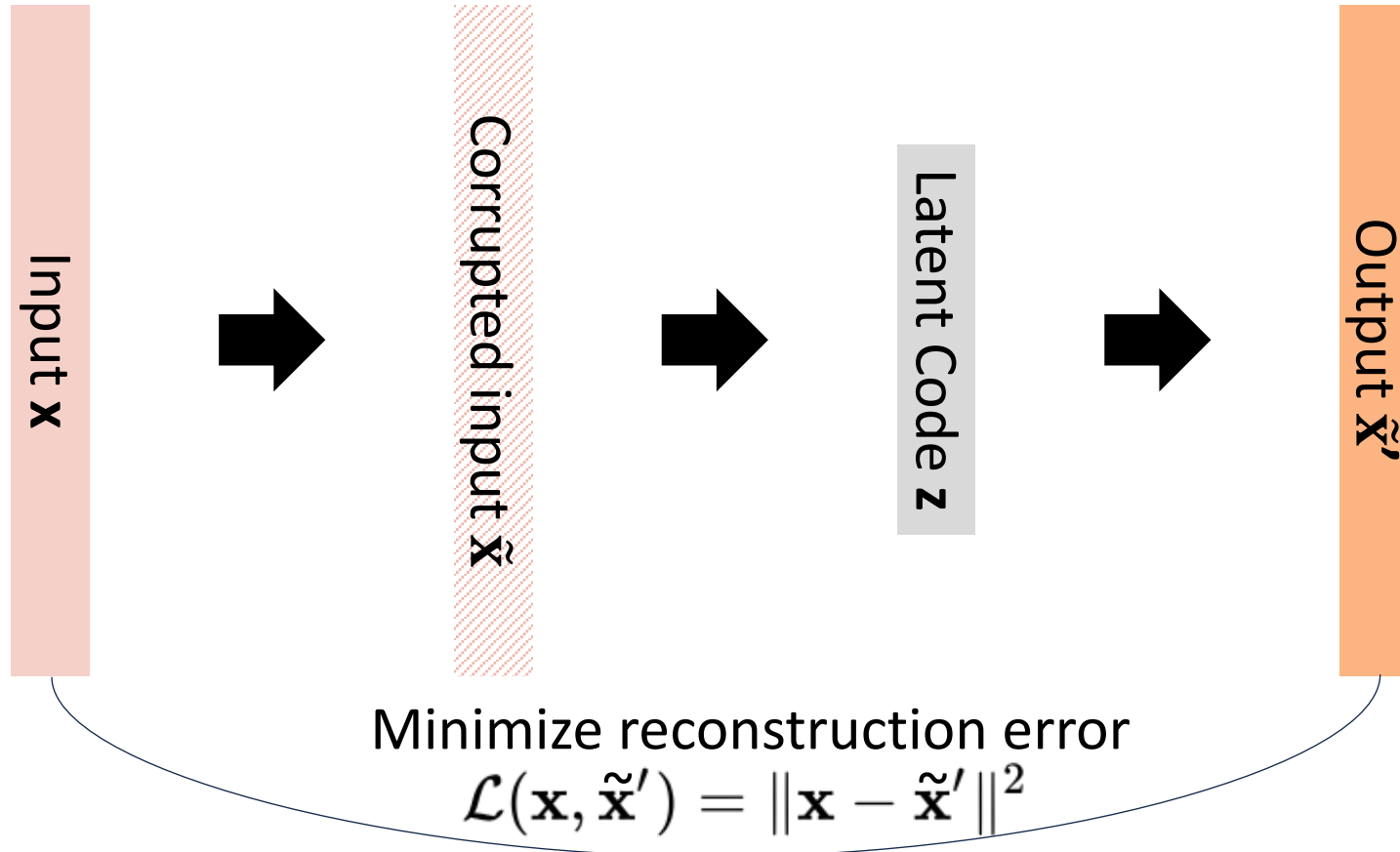
- Corrupt the input sample  $\mathbf{x}$



Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders, ICML' 08

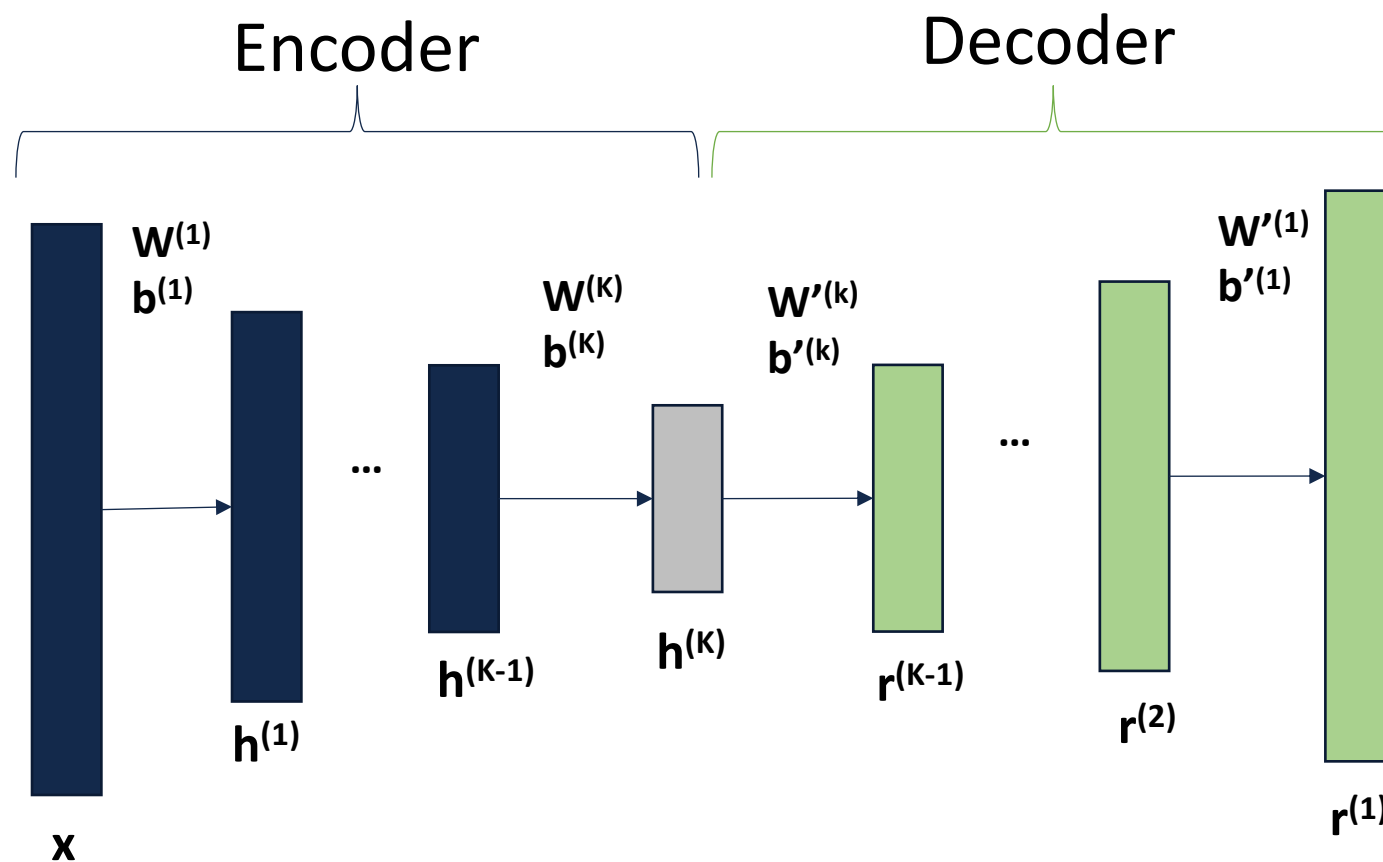
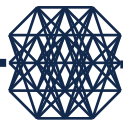
# Denoising Autoencoders

- Try to reconstruct the original uncorrupted input  $\mathbf{x}$



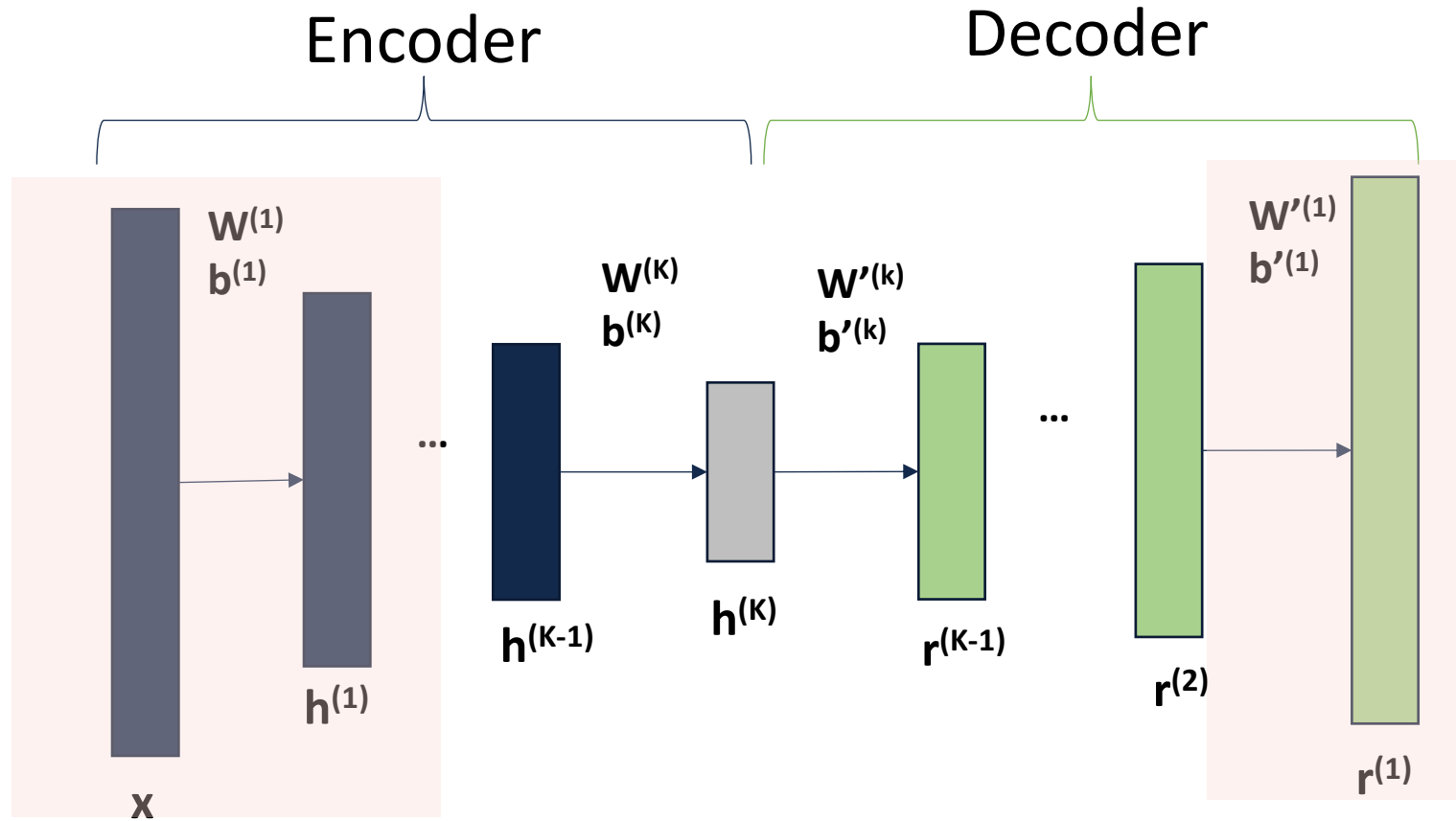


# Stacked Autoencoder



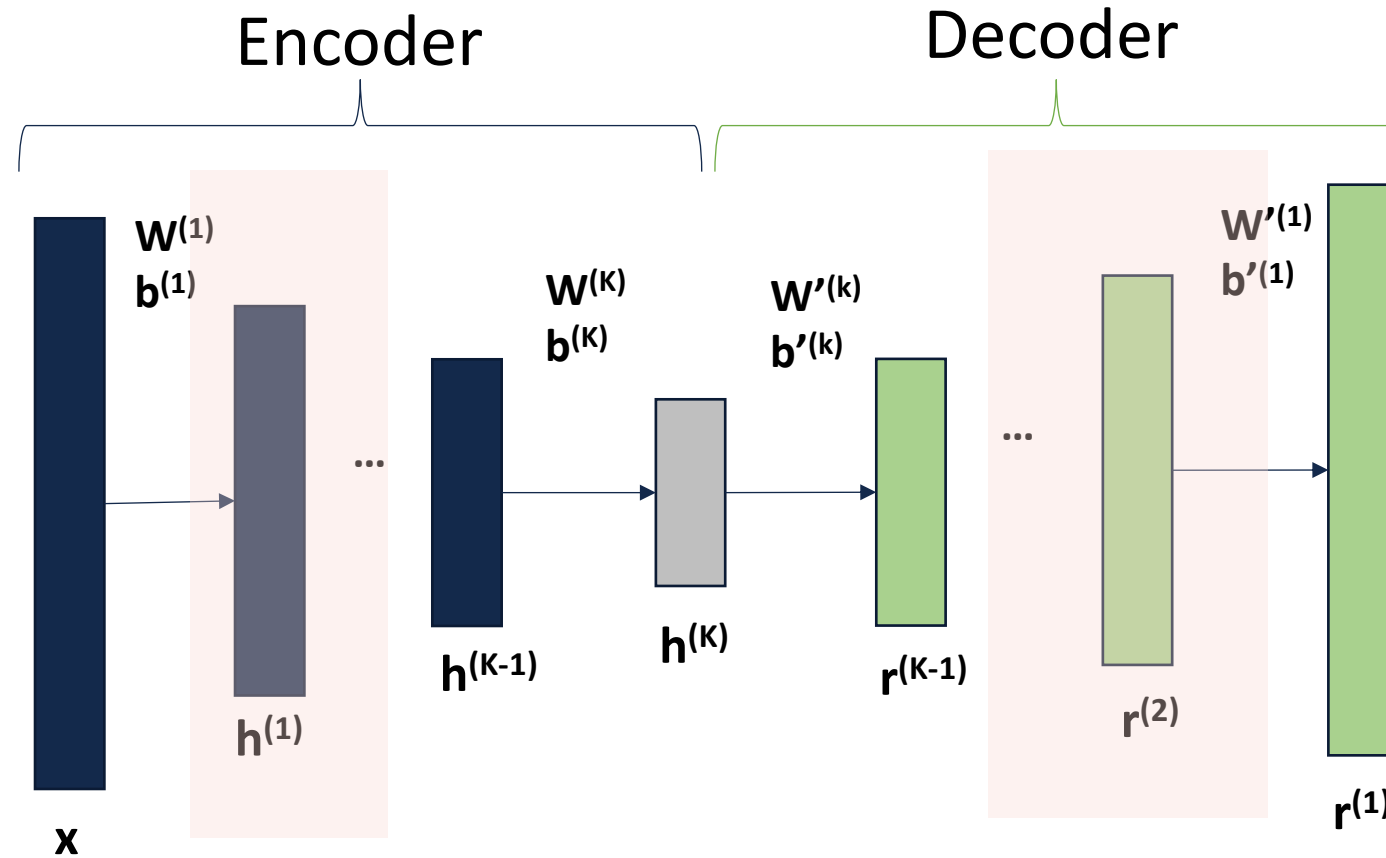
- Stacked multiple autoencoders together

# Stacked Autoencoder: layer-wise pretraining



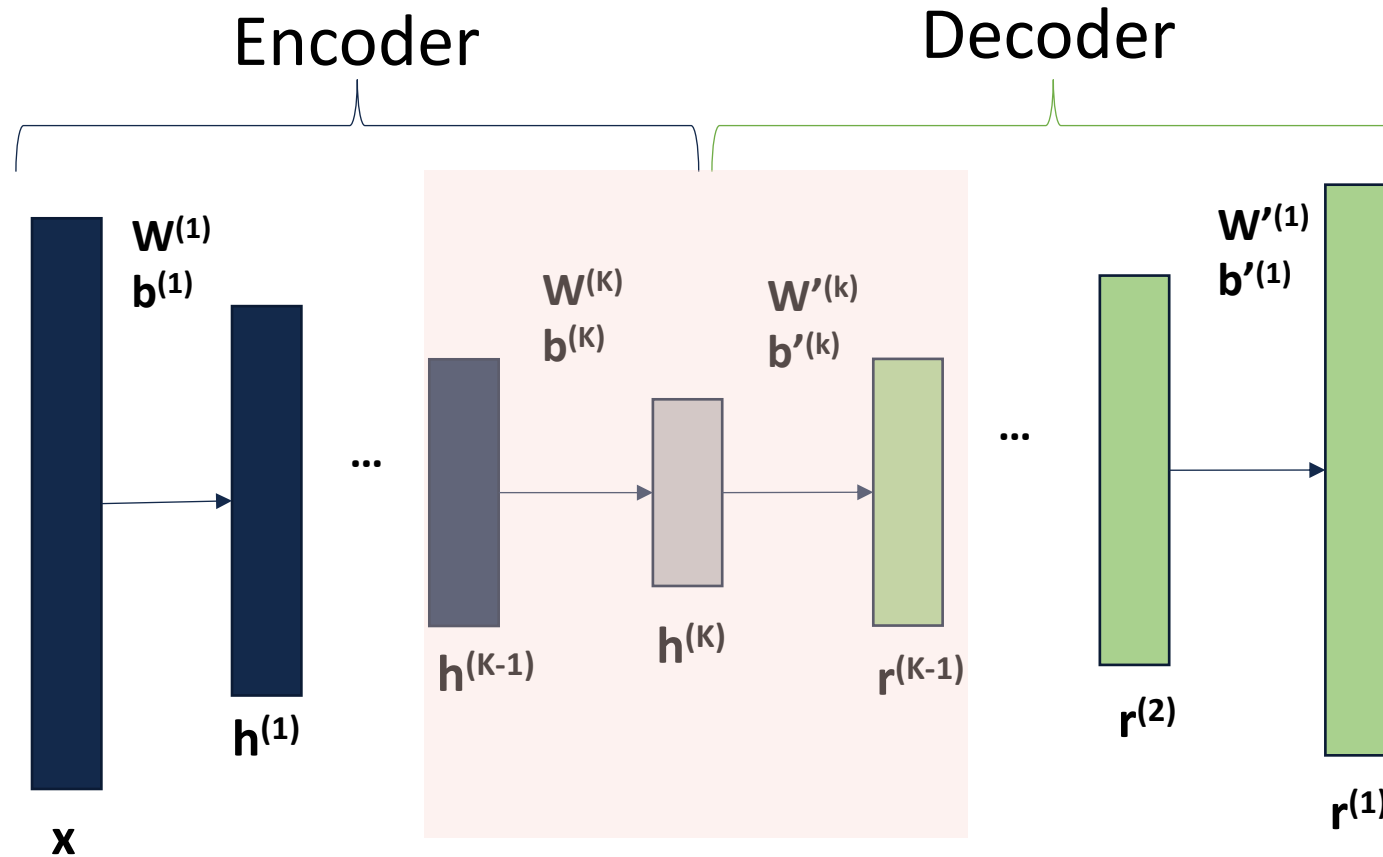
- Layer-wise training

# Stacked Autoencoder: layer-wise pretraining



- Layer-wise training

# Stacked Autoencoder: layer-wise pretraining



- Layer-wise training

# Computational Phenotype Discovery Using Unsupervised Feature Learning over Noisy, Sparse, and Irregular Clinical Data

Thomas A. Lasko , Joshua C. Denny, Mia A. Levy

Lasko, Thomas A., Joshua C. Denny, and Mia A. Levy. 2013. "Computational Phenotype Discovery Using Unsupervised Feature Learning over Noisy, Sparse, and Irregular Clinical Data." *PloS One* 8 (6):e66341.

# Computational Phenotype Discovery



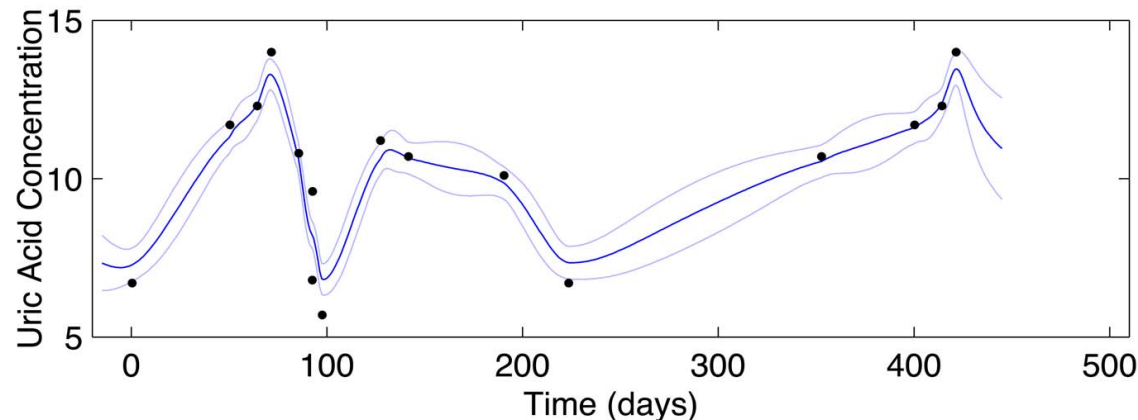
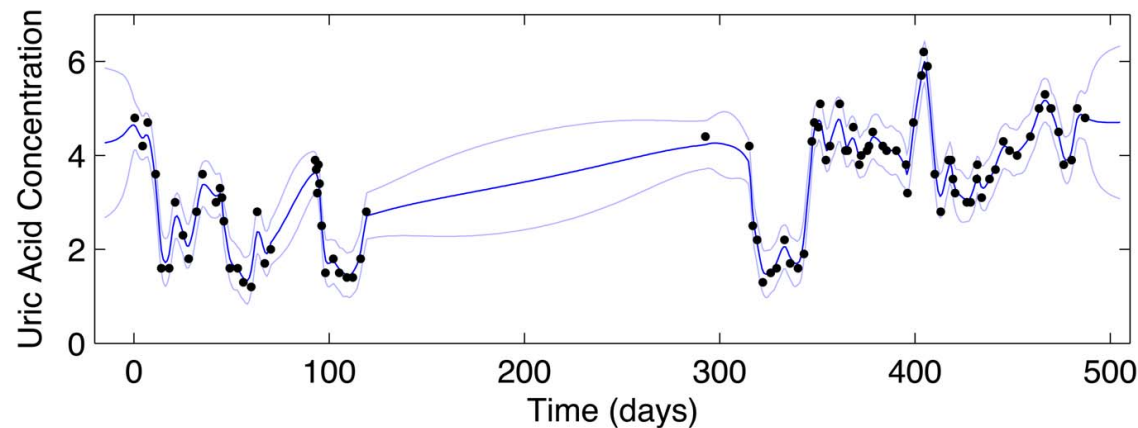
Computational phenotype discovery using unsupervised feature learning over noisy, sparse, and irregular clinical data

Lasko, Thomas A., Joshua C. Denny, and Mia A. Levy, PloS one 2013

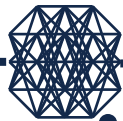
- Learn phenotypic patterns from EHRs without expert knowledge
  - Learn temporal **serum uric acid** patterns to classify gout VS acute leukemia

# Impute missing data

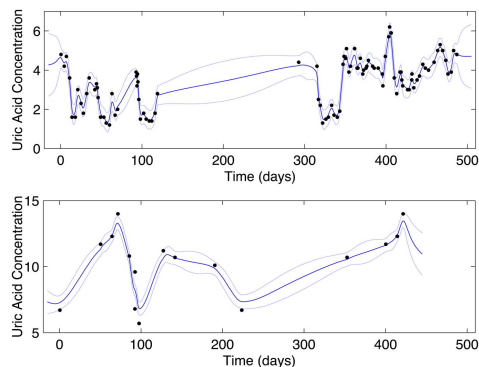
- Gaussian process regression is used to convert discrete observation into continuous estimate



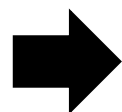
# Phenotype Discovery



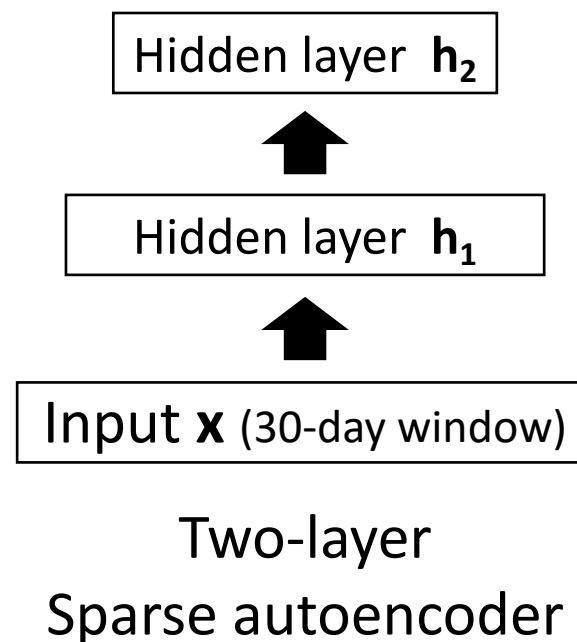
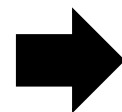
- Modeling pipeline



Gaussian Process  
Regression



$\mathbf{m}, \mathbf{s}$   
Sequence of means &  
standard deviations



Loss function  $L_N(\mathbf{m}, \hat{\mathbf{m}}, \mathbf{s}) = \sum_{i=1}^M \left[ \frac{\hat{m}_i - m_i}{s_i} \right]^2$



# Experiment setup



- Data: 4368 serum uric acid time-series from Vanderbilt

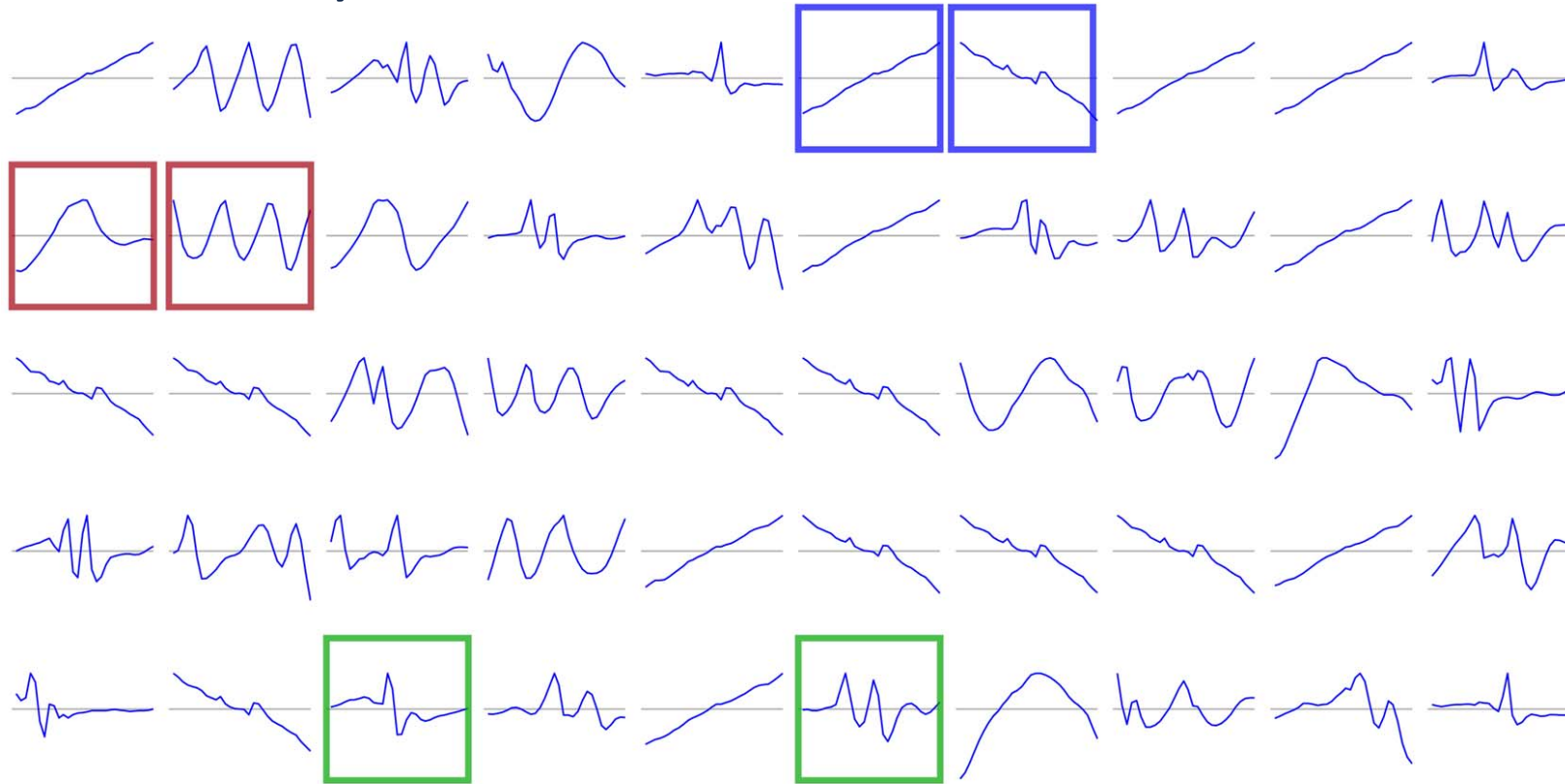
**Table 1.** Statistical characteristics of uric acid sequences in gout vs. leukemia.

Attribute	Gout	Leukemia
Number of Sequences	2194	2174
Minimum	0.9	0.0
1st Quartile	6.2	3.0
Median	7.7	4.2
3rd Quartile	9.5	5.6
Maximum	34.0	75.0

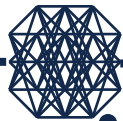
# Phenotype Patterns



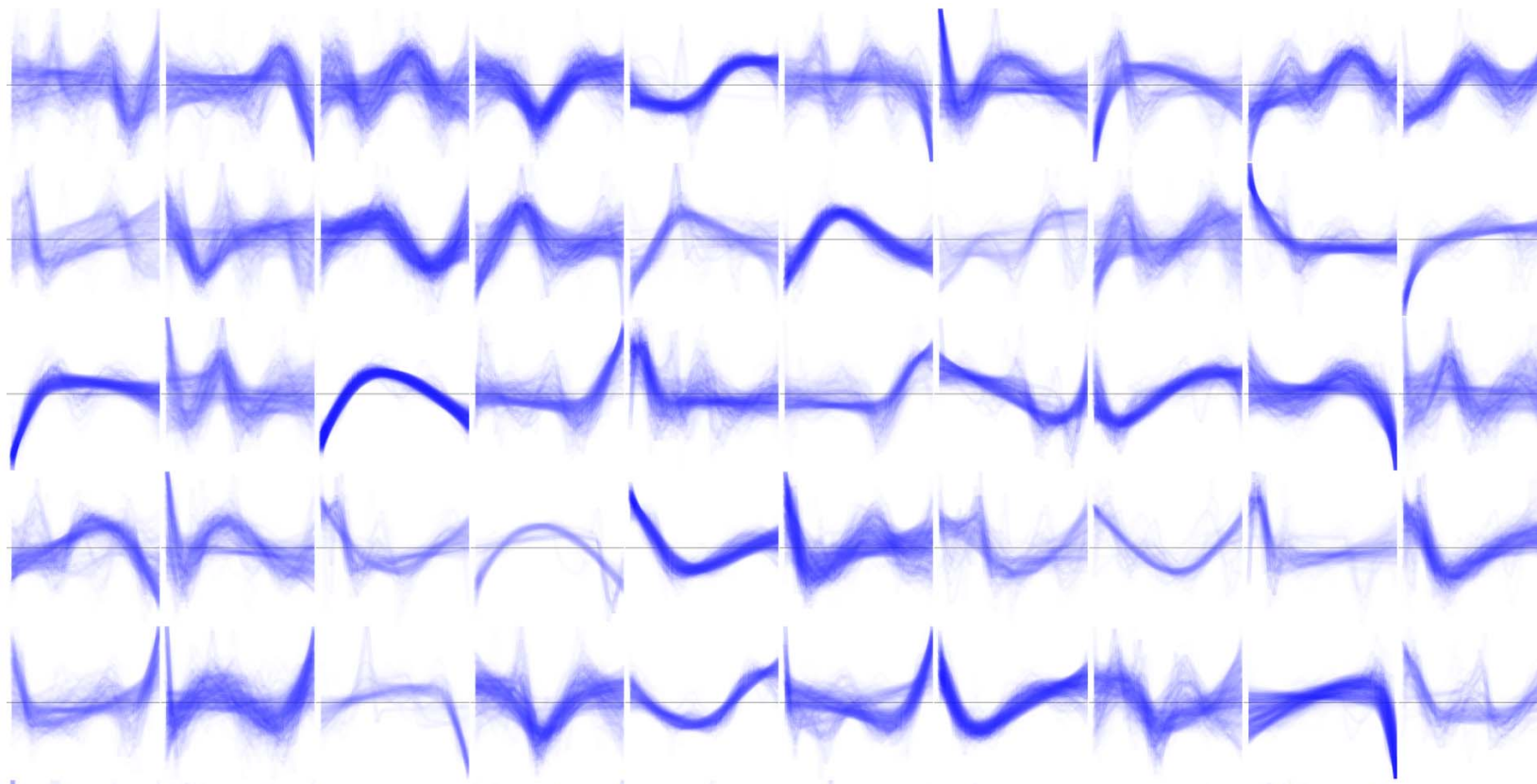
- Patterns at the 1<sup>st</sup> layer of the sAE



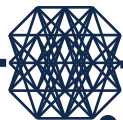
# Computational Phenotype Discovery



- Patterns at the 2<sup>nd</sup> layer of the sAE



# Classification performance



- Accurate classification can be achieved using hidden layers from autoencoder

**Table 3.** Unsupervised features were as powerful as expert-engineered features in distinguishing uric acid sequences from gout vs. leukemia.

Classifier	AUC (training)	AUC [CI] (test)
First-Layer Learned Features	0.969	0.972 [0.968, 0.979]
Second-Layer Learned Features	0.965	0.972 [0.968, 0.979]
Expert Engineered Features	0.968	0.974 [0.966, 0.981]
Baseline (sequence mean only)	0.922	0.932 [0.922, 0.944]

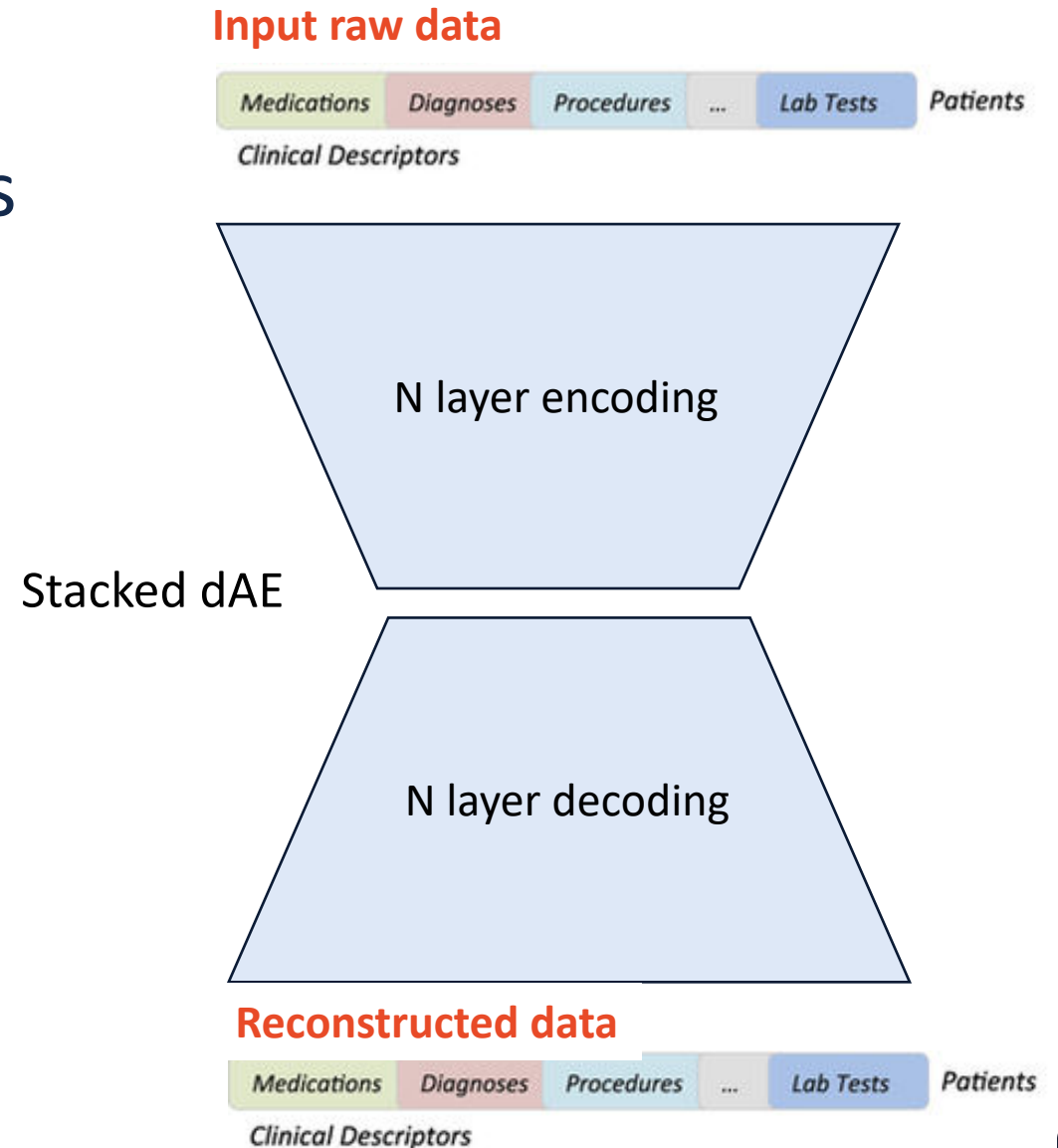
# Deep Patient: An Unsupervised Representation to Predict the Future of Patients from the Electronic Health Records

Riccardo Miotto, Li Li, Brian A. Kidd & Joel T. Dudley

Miotto, Riccardo, Li Li, Brian A. Kidd, and Joel T. Dudley. 2016. "Deep Patient: An Unsupervised Representation to Predict the Future of Patients from the Electronic Health Records." *Scientific Reports* 6 (May):26094.

# Deep Patient

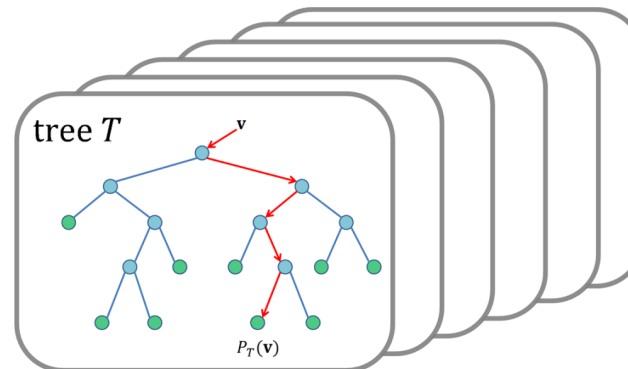
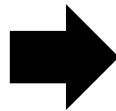
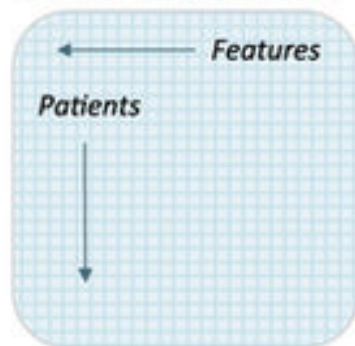
- Learn unsupervised representations of patients for general predictive healthcare
- Multiple denoising autoencoders (dAE) are stacked together to form the deep learning model



# Deep Patient: Experiment

- Data
  - From 700K patients from Mount Sinai hospital
  - Use multiple data modalities
    - Diagnosis/medication/procedure codes, lab tests, clinical notes, demographic info
  - Output labels
    - 78 diagnosis codes
- use sAE features to train a random forest to classify 78 diagnosis codes

Deep Patient Dataset



# Deep Patient: Results in AUC

Time Interval = 1 year (76,214 patients)			
Disease	Area under the ROC curve		
	RawFeat	PCA	DeepPatient
Diabetes mellitus with complications	0.794	0.861	<b>0.907</b>
Cancer of rectum and anus	0.863	0.821	<b>0.887</b>
Cancer of liver and intrahepatic bile duct	0.830	0.867	<b>0.886</b>
Regional enteritis and ulcerative colitis	0.814	0.843	<b>0.870</b>
Congestive heart failure (non-hypertensive)	0.808	0.808	<b>0.865</b>
Attention-deficit and disruptive behavior disorders	0.730	0.797	<b>0.863</b>
Cancer of prostate	0.692	0.820	<b>0.859</b>
Schizophrenia	0.791	0.788	<b>0.853</b>
Multiple myeloma	0.783	0.739	<b>0.849</b>
Acute myocardial infarction	0.771	0.775	<b>0.847</b>



# Outro

- Autoencoder
- Variants of autoencoders
  - Sparse autoencoder
  - Denoising autoencoder
  - Stacked autoencoder
- Healthcare applications of autoencoders