# From Shallow to Deep Kernel Machines

**Johan Suykens**

KU Leuven, ESAT-Stadius and Leuven.AI Institute
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
Email: johan.suykens@esat.kuleuven.be
http://www.esat.kuleuven.be/stadius/

**ICINCO 2026, Angers France, Oct 2026**



# Overview

- Introduction and Motivation
- Least Squares Support Vector Machines (LS-SVM) as core models
- LS-SVM and Deep Learning:

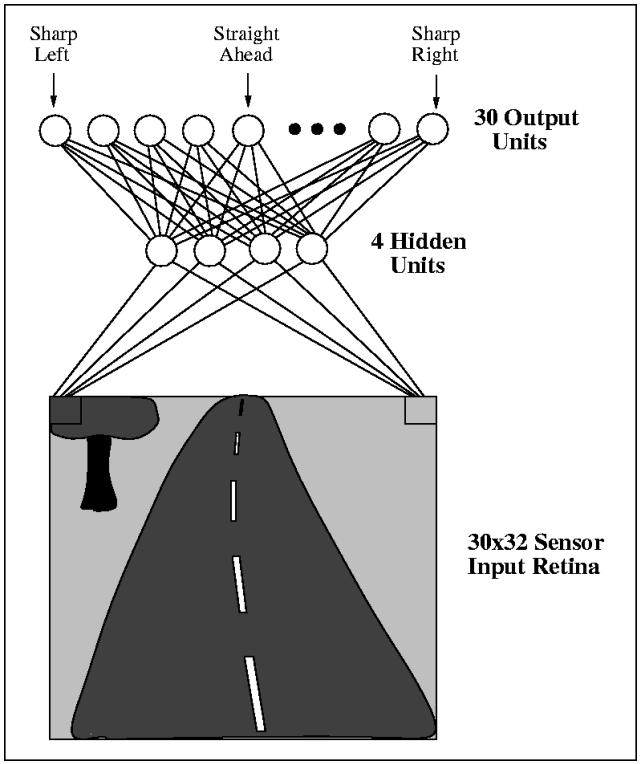
  $\rightarrow$ From Restricted Kernel Machines (RKM) to Deep Kernel Machines

  $\rightarrow$ Kernel SVD, Self-Attention and Transformers
- Future challenges
- Conclusions

# *Introduction*

# Neural networks as universal approximators

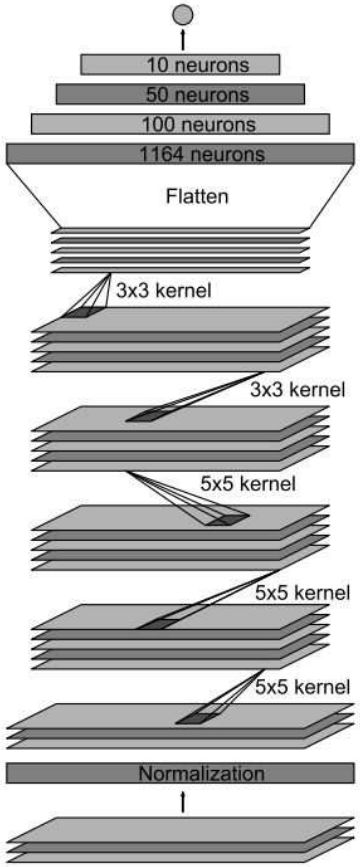
One hidden layer is sufficient for universal approximation:






ALVINN (Autonomous Land Vehicle In a Neural Network)
[Pomerleau, Neural Computation 1991]

# Deep feature learning




Output: vehicle control

Fully-connected layer
Fully-connected layer
Fully-connected layer

Convolutional feature map 64@1x18

Convolutional feature map 64@3x20

Convolutional feature map 48@5x22

Convolutional feature map 36@14x47

Convolutional feature map 24@31x98

Normalized input planes 3@66x200

Input planes 3@66x200

(27 million connections)



Waymo / Google Self-Driving Car



Tesla Autopilot




Uber




nuTonomy

from: [selfdrivingcars.mit.edu (Lex Fridman et al.), 2017]

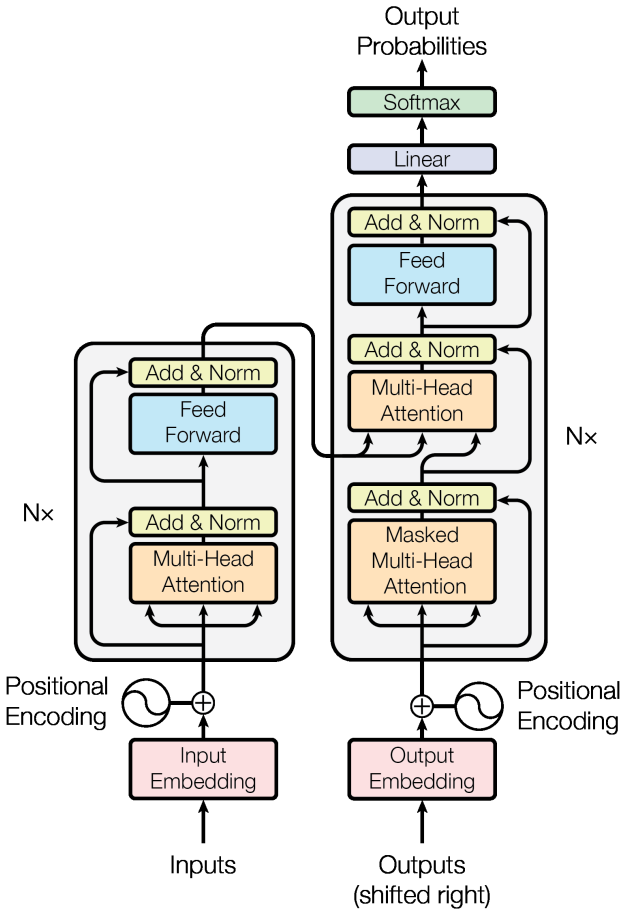
# Generative models

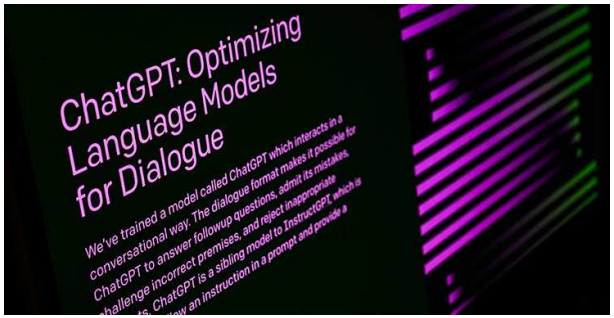



**Fig. 13.1** Four years of face generation using generative models

[figure Ye 2022]

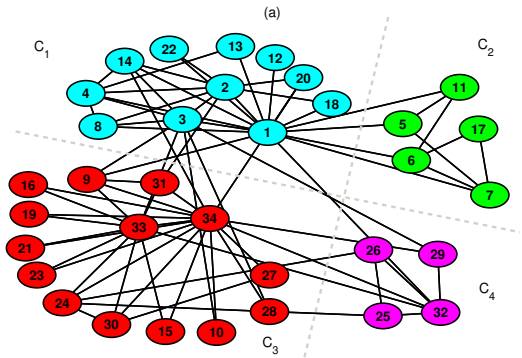
# Transformers



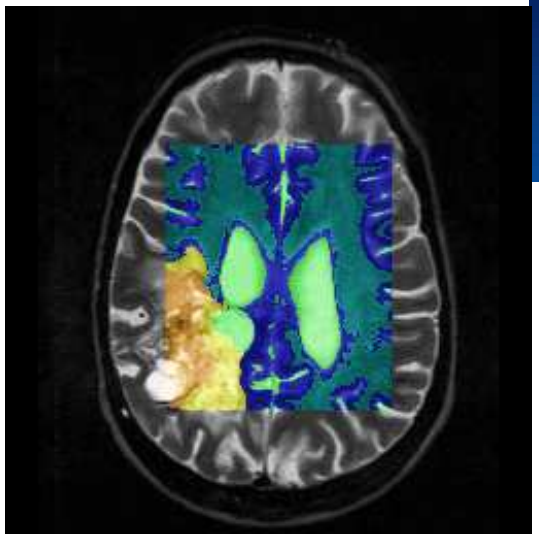
# Data-driven world

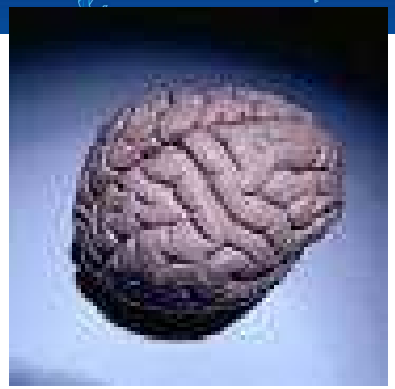








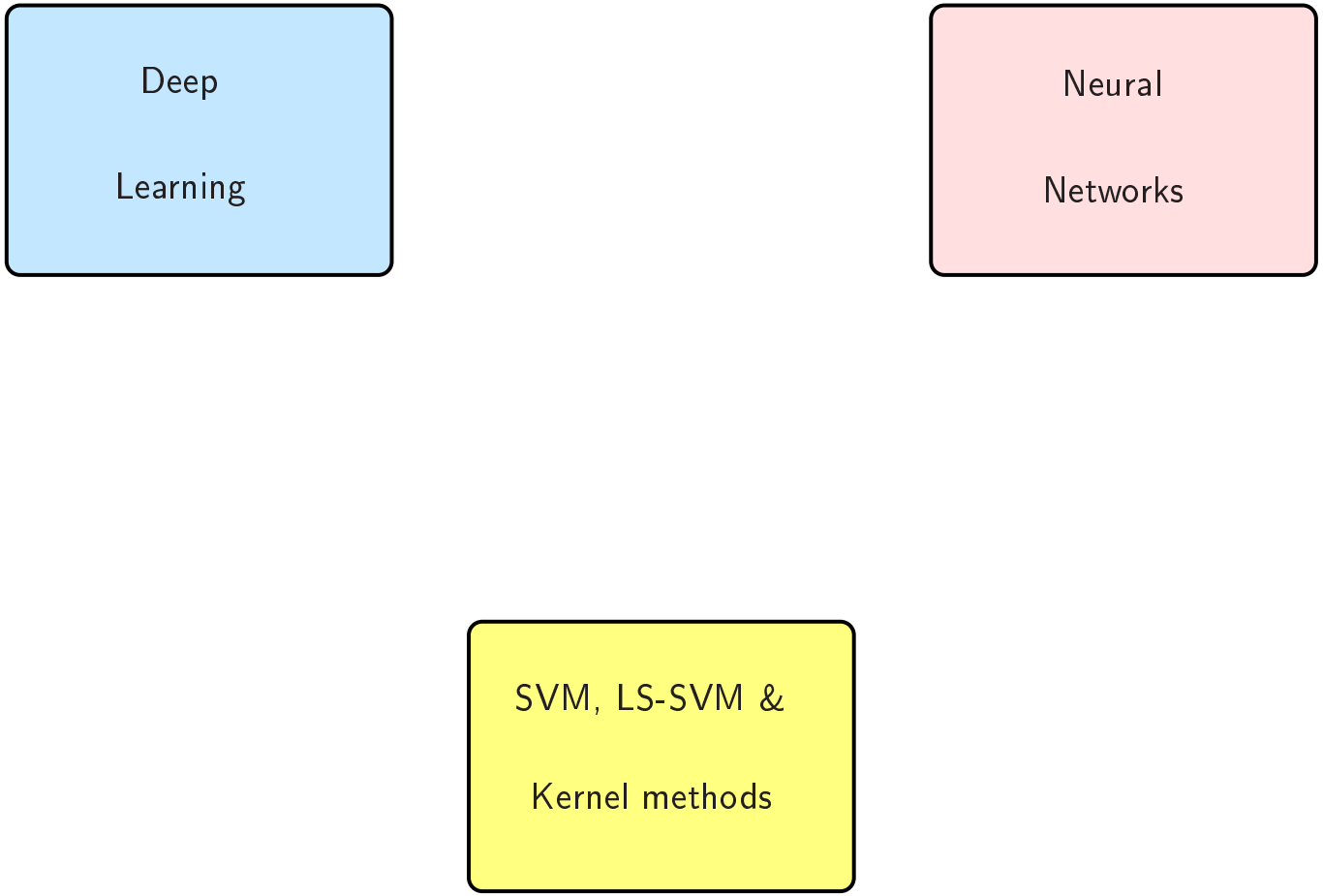




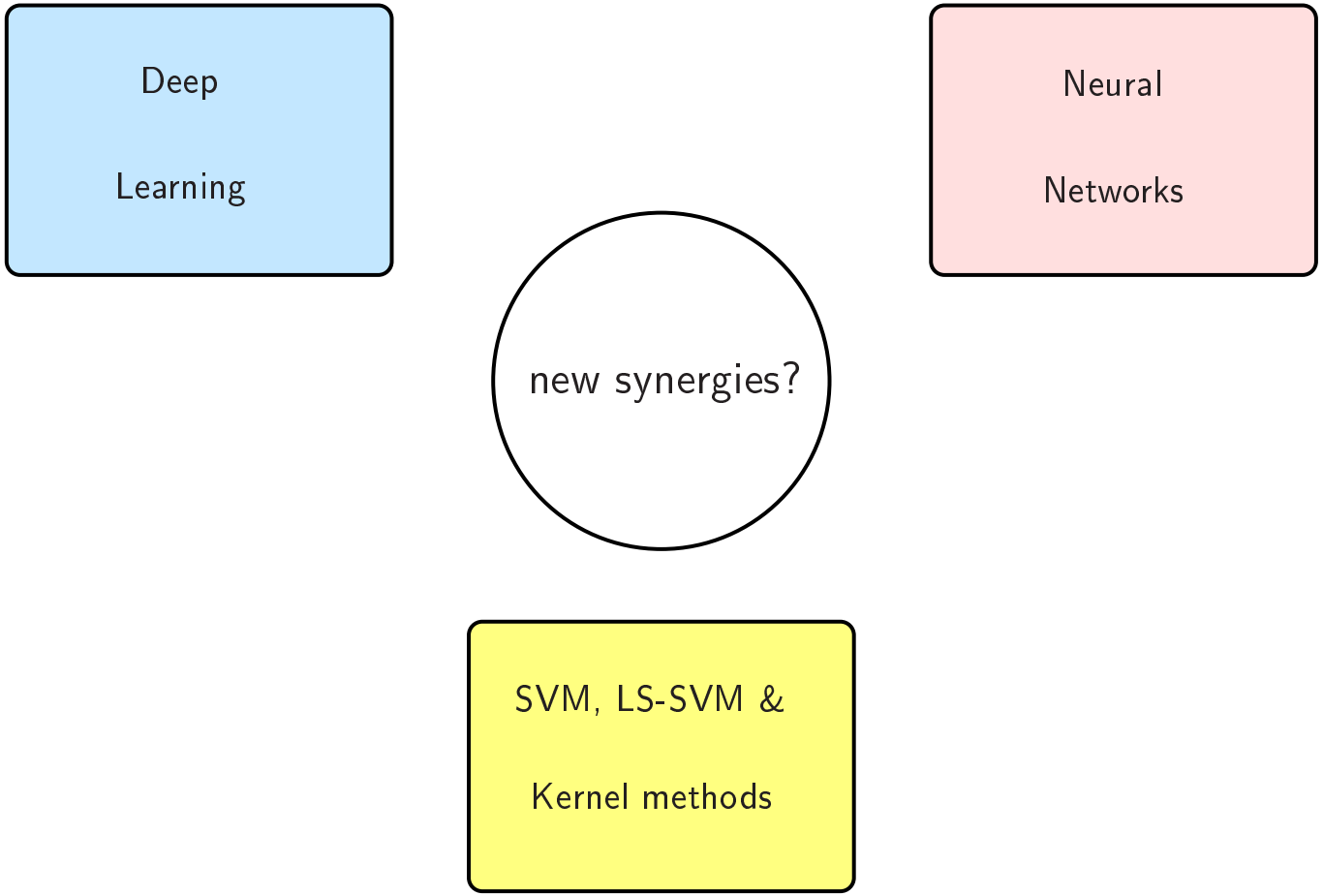
# Challenges

- general methodology
- unifying frameworks
- mathematical foundations
- scalability
- robustness
- interpretability

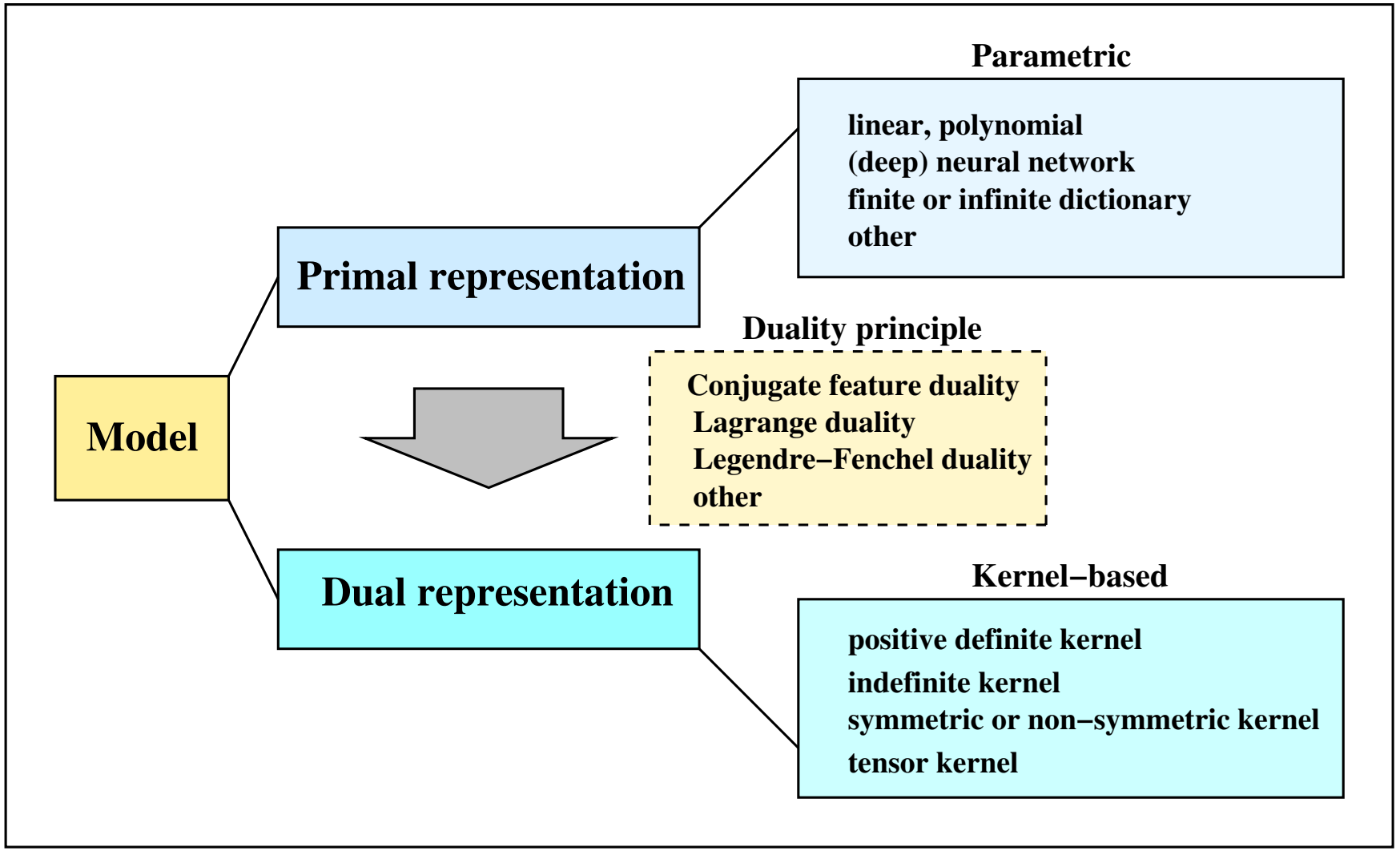
# New Synergies



# New Synergies



# Towards a unifying picture




[Suykens 2017]

# *LS-SVM as core models*

# SVM and LS-SVM classifier

- **SVM Primal problem:** [Vapnik, 1995; Cortes & Vapnik, 1995]

$$\min_{w,b,\xi} \ \frac{1}{2}w^T w + c\sum_{i=1}^{N}\xi_i \ \ \text{s.t.} \ \ \begin{cases} y_i[w^T\varphi(x_i)+b] \geq 1-\xi_i \\ \xi_i \geq 0, \quad i=1,...,N \end{cases}$$

  Dual problem: convex QP problem

- **LS-SVM Primal problem:** [Suykens & Vandewalle, 1999]

$$\min_{w,b,e} \ \frac{1}{2}w^T w + \frac{1}{2}\gamma\sum_{i=1}^{N}e_i^2 \ \ \text{s.t.} \quad y_i[w^T\varphi(x_i)+b] = 1-e_i$$

  Dual problem: linear system of equations

# Feature map and kernel

Primal and dual representation:

$$\text{Model} \begin{cases} \nearrow & (P): \quad \hat{y} = w^T \varphi(x) + b \\ \searrow & (D): \quad \hat{y} = \sum_i \alpha_i K(x_i, x) + b \end{cases}$$

Mercer theorem (one can **either** choose $\varphi$ **or** positive definite $K$):

$$K(x, z) = \varphi(x)^T \varphi(z)$$

Feature map $\varphi$, Kernel function $K(x, z)$ (e.g. linear, polynomial, RBF, ...)

- SVMs: feature map and positive definite kernel [Cortes & Vapnik, 1995]
- Neural networks: hidden layer as feature map [Suykens & Vandewalle, IEEE-TNN 1999]
- Least squares support vector machines [Suykens et al., 2002]: $L_2$ loss and regularization

# Least Squares Support Vector Machines: "core models"

- Regression

$$\min_{w,b,e} w^T w + \gamma \sum_i e_i^2 \quad \text{s.t.} \quad y_i = w^T \varphi(x_i) + b + e_i, \quad \forall i$$

- Classification

$$\min_{w,b,e} w^T w + \gamma \sum_i e_i^2 \quad \text{s.t.} \quad y_i (w^T \varphi(x_i) + b) = 1 - e_i, \quad \forall i$$

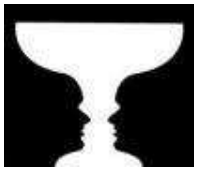
- Kernel pca $(V = I)$, Kernel spectral clustering $(V = D^{-1})$

$$\min_{w,b,e} -w^T w + \gamma \sum_i v_i e_i^2 \quad \text{s.t.} \quad e_i = w^T \varphi(x_i) + b, \quad \forall i$$

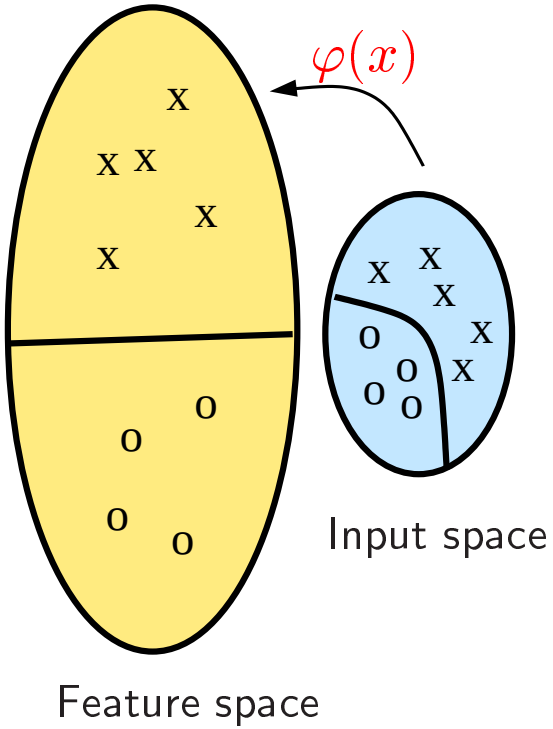
- Kernel canonical correlation analysis/partial least squares

$$\min_{w,v,b,d,e,r} w^T w + v^T v + \nu \sum_i (e_i - r_i)^2 \text{ s.t. } \begin{cases} e_i &= w^T \varphi^{(1)}(x_i) + b \\ r_i &= v^T \varphi^{(2)}(y_i) + d \end{cases}$$

[Suykens & Vandewalle, NPL 1999; Suykens et al., 2002; Alzate & Suykens, 2010]



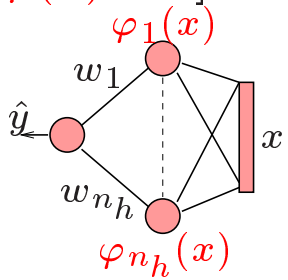
# SVMs and neural networks




## Primal space

Parametric

$$\hat{y} = \text{sign}[w^T \varphi(x) + b]$$



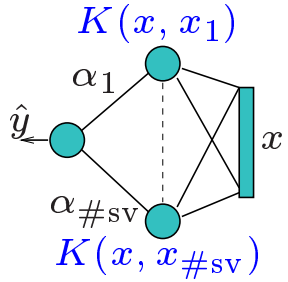

Parametric

$K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$ (“Kernel trick”)

## Dual space

Nonparametric

$$\hat{y} = \text{sign}[\sum_{i=1}^{\#\text{sv}} \alpha_i y_i K(x, x_i) + b]$$




Non–parametric

[Suykens et al., 2002]

# Kernels

Wide range of positive definite kernel functions possible:

- linear $K(x, z) = x^T z$
- polynomial $K(x, z) = (\eta + x^T z)^d$
- radial basis function $K(x, z) = \exp(-\|x - z\|_2^2/\sigma^2)$
- splines, wavelets
- string kernel, Fisher kernel
- graph kernels, kernels from graphical models
- kernels for dynamical systems
- data fusion kernels
- other

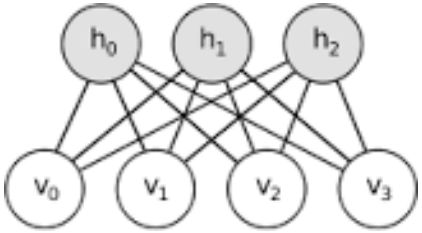
[Schölkopf & Smola, 2002; Shawe-Taylor & Cristianini, 2004; Jebara et al., 2004; other]

Function estimation in RKHS, RKKS, RKBS; Gaussian processes

[Wahba 1990; Steinwart & Christmann 2008; Cucker & Zhou 2007; Rasmussen & Williams 2006]

*From LS-SVM framework to RKM representations, and beyond*

# Restricted Boltzmann Machines (RBM)




- Markov random field, bipartite graph, stochastic binary units
  Layer of visible units $v$ and layer of hidden units $h$
  **No hidden-to-hidden connections**
- Energy:

$$E(v, h; \theta) = -v^T W h - c^T v - a^T h \quad \text{with} \quad \theta = \{W, c, a\}$$

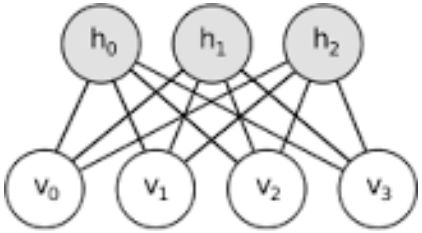
Joint distribution:

$$P(v, h; \theta) = \frac{1}{Z(\theta)} \exp(-E(v, h; \theta))$$

with partition function $Z(\theta) = \sum_v \sum_h \exp(-E(v, h; \theta))$

[Hinton, Osindero, Teh, Neural Computation 2006]

# Restricted Boltzmann Machines (RBM)




- Markov random field, bipartite graph, stochastic binary units
  Layer of visible units $v$ and layer of hidden units $h$
  **No hidden-to-hidden connections**
- Energy:

$$E(v, h; \theta) = -v^T W h - c^T v - a^T h \quad \text{with} \quad \theta = \{W, c, a\}$$

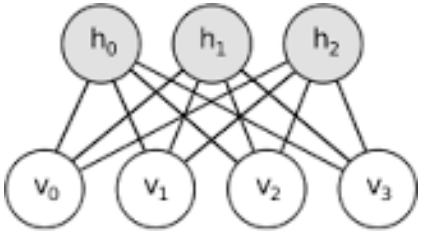
Joint distribution:

$$P(v, h; \theta) = \frac{1}{Z(\theta)} \exp(-E(v, h; \theta))$$

with partition function $Z(\theta) = \sum_v \sum_h \exp(-E(v, h; \theta))$

[Hinton, Osindero, Teh, Neural Computation 2006]

# Restricted Boltzmann Machines (RBM)




- Markov random field, bipartite graph, stochastic binary units
  Layer of visible units $v$ and layer of hidden units $h$
  **No hidden-to-hidden connections**
- Energy:

$$E(v,h;\theta) = -(v^T W)h - c^T v - a^T h \quad \text{with} \quad \theta = \{W, c, a\}$$

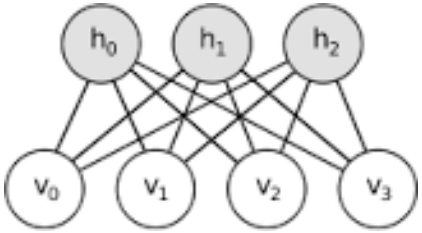
Joint distribution:

$$P(v,h;\theta) = \frac{1}{Z(\theta)} \exp(-E(v,h;\theta))$$

with partition function $Z(\theta) = \sum_v \sum_h \exp(-E(v,h;\theta))$

[Hinton, Osindero, Teh, Neural Computation 2006]

# Restricted Boltzmann Machines (RBM)




- Markov random field, bipartite graph, stochastic binary units
  Layer of visible units $v$ and layer of hidden units $h$
  **No hidden-to-hidden connections**
- Energy:

$$E(v, h; \theta) = -v^T(Wh) - c^T v - a^T h \quad \text{with} \quad \theta = \{W, c, a\}$$

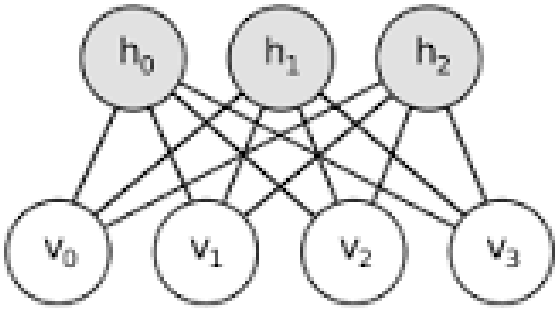
Joint distribution:

$$P(v, h; \theta) = \frac{1}{Z(\theta)} \exp(-E(v, h; \theta))$$

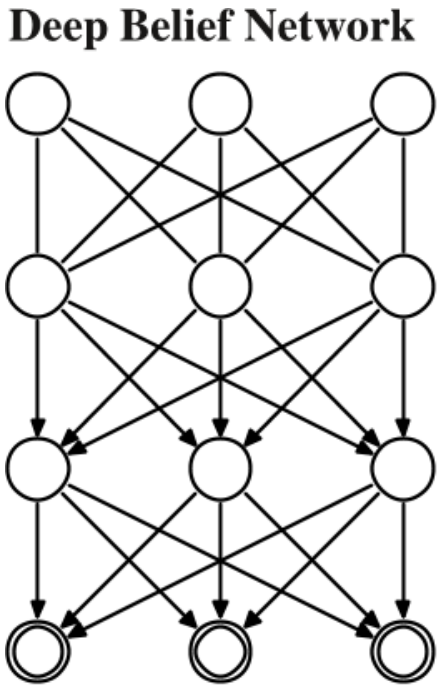
with partition function $Z(\theta) = \sum_v \sum_h \exp(-E(v, h; \theta))$

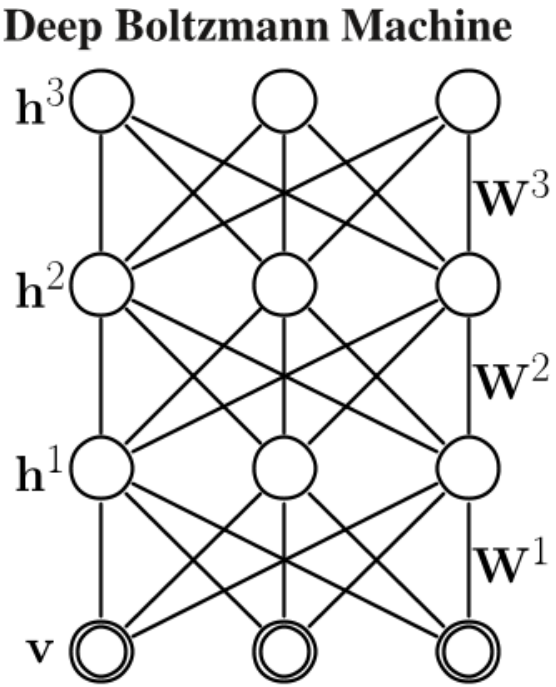
[Hinton, Osindero, Teh, Neural Computation 2006]

# RBM and deep learning




$p(v, h)$



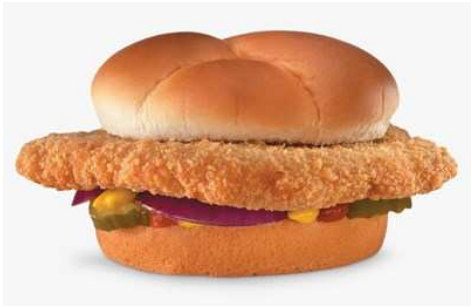




$p(v, h^1, h^2, h^3, ...)$

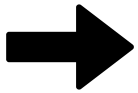
[Hinton et al., 2006; Salakhutdinov, 2015]

# in other words ...

*"sandwich"*



$$E = -v^T W h$$

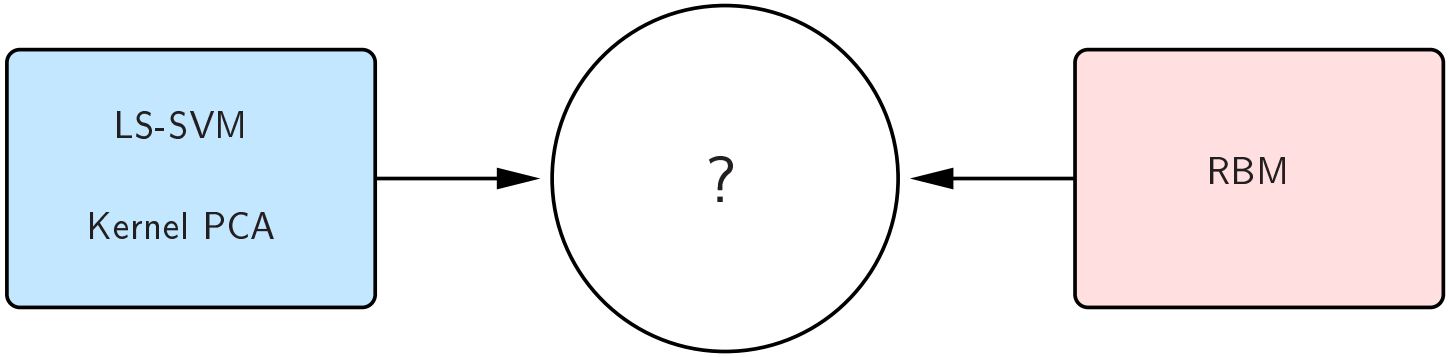


*"deep sandwich"*

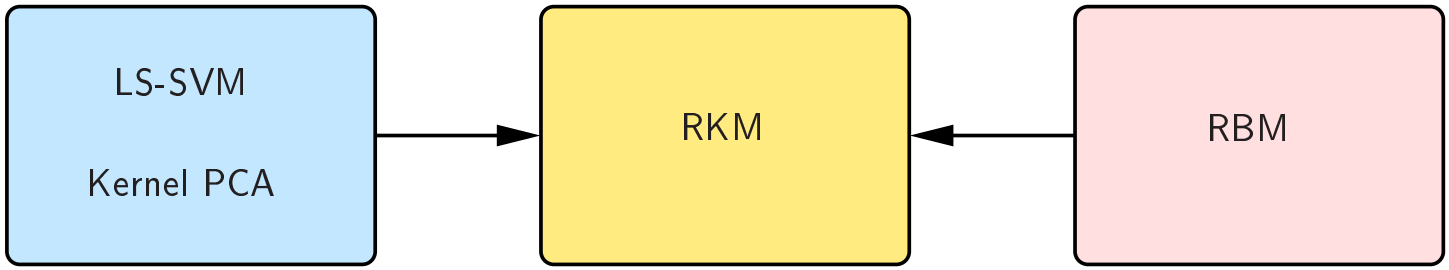


$$E = -v^T W^1 h^1 - {h^1}^T W^2 h^2 - {h^2}^T W^3 h^3$$

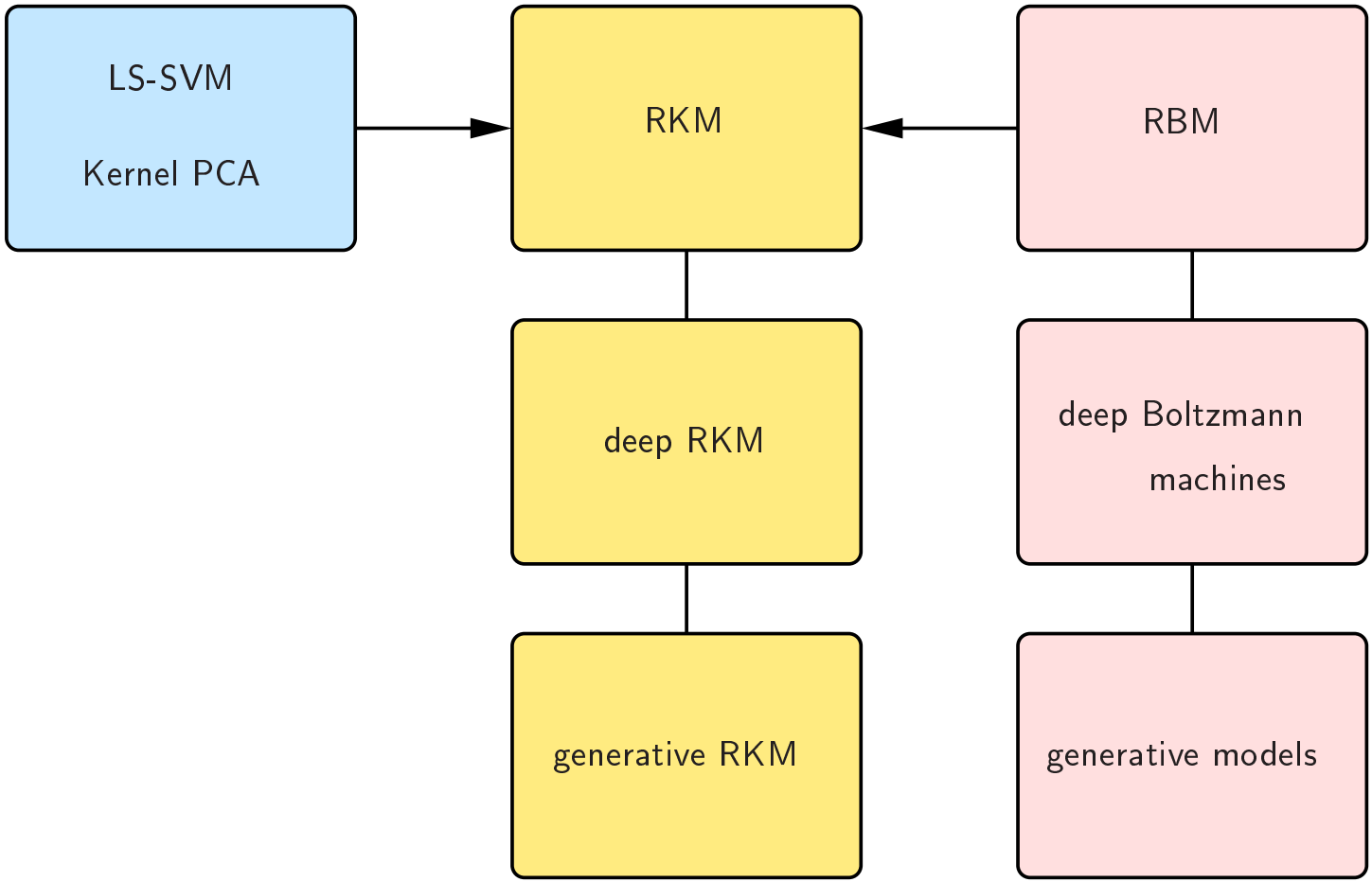
# Connecting LSSVM/KPCA & RBM



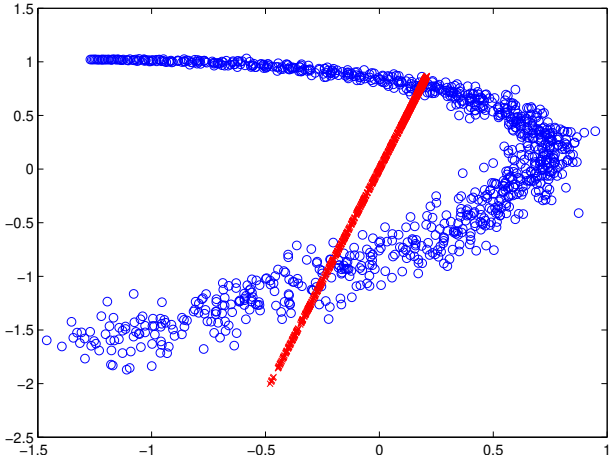
# Connecting LSSVM/KPCA & RBM



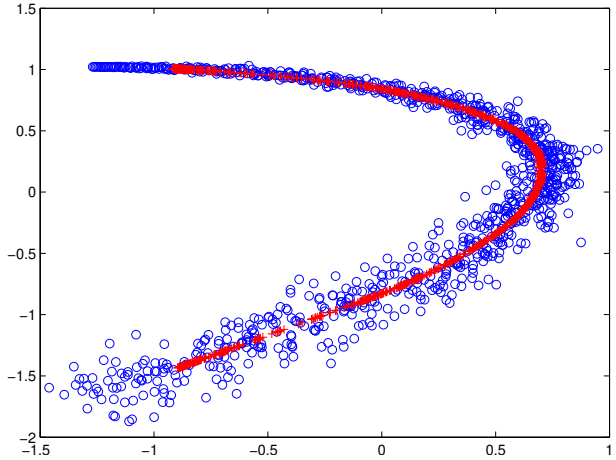
# Connecting LSSVM/KPCA & RBM



# Kernel principal component analysis (KPCA)




*linear PCA*




*kernel PCA (RBF kernel)*

**Kernel PCA** [Schölkopf et al., 1998]:
take eigenvalue decomposition of the kernel matrix

$$\begin{bmatrix} K(x_1, x_1) & \dots & K(x_1, x_N) \\ \vdots & & \vdots \\ K(x_N, x_1) & \dots & K(x_N, x_N) \end{bmatrix}$$

(applications in dimensionality reduction and denoising)

# Kernel PCA: classical LS-SVM approach

- Primal problem: [Suykens et al., 2002]: **model-based** approach

$$\min_{w,b,e} \ \frac{1}{2}w^T w - \frac{1}{2}\gamma \sum_{i=1}^{N} e_i^2 \ \text{ s.t. } \ e_i = w^T \varphi(x_i) + b, \ i = 1, ..., N.$$

- Dual problem (Lagrange duality) corresponds to kernel PCA

$$\Omega^{(c)}\alpha = \lambda\alpha \ \text{ with } \ \lambda = 1/\gamma$$

with $\Omega_{ij}^{(c)} = (\varphi(x_i) - \hat{\mu}_\varphi)^T(\varphi(x_j) - \hat{\mu}_\varphi)$ the *centered kernel matrix* and $\hat{\mu}_\varphi = (1/N)\sum_{i=1}^{N}\varphi(x_i)$.

- Interpretation:
  1. pool of candidate components (objective function equals zero)
  2. select relevant components

# From KPCA to RKM representation (1)



Model:

$e = W^T \varphi(x)$

objective $J$
$=$ regularization term $\mathrm{Tr}(W^T W)$
- $(\frac{1}{\lambda})$ variance term $\sum_i e_i^T e_i$

$\downarrow$ use property $e^T h \leq \frac{1}{2\lambda} e^T e + \frac{\lambda}{2} h^T h$

RKM representation:

$e = \sum_j h_j K(x_j, x)$

obtain $J \leq \overline{J}(h_i, W)$
solution from stationary points of $\overline{J}$:
$\frac{\partial \overline{J}}{\partial h_i} = 0$, $\frac{\partial \overline{J}}{\partial W} = 0$

# From KPCA to RKM representation (2)

- Objective [Suykens, Neural Computation 2017]

$$
\begin{aligned}
J &= \frac{\eta}{2}\mathrm{Tr}(W^TW) - \frac{1}{2\lambda}\sum_{i=1}^{N} e_i^T e_i \ \text{ s.t. } e_i = W^T\varphi(x_i), \forall i \\
&\leq -\sum_{i=1}^{N} e_i^T h_i + \frac{\lambda}{2}\sum_{i=1}^{N} h_i^T h_i + \frac{\eta}{2}\mathrm{Tr}(W^TW) \ \text{ s.t. } e_i = W^T\varphi(x_i), \forall i \\
&= -\sum_{i=1}^{N} \varphi(x_i)^T W h_i + \frac{\lambda}{2}\sum_{i=1}^{N} h_i^T h_i + \frac{\eta}{2}\mathrm{Tr}(W^TW) \triangleq \overline{J}
\end{aligned}
$$

- Stationary points of $\overline{J}(h_i, W)$:

$$
\begin{cases}
\dfrac{\partial \overline{J}}{\partial h_i} = 0 \quad \Rightarrow \quad W^T\varphi(x_i) = \lambda h_i, \ \forall i \\
\dfrac{\partial \overline{J}}{\partial W} = 0 \quad \Rightarrow \quad W = \dfrac{1}{\eta}\sum_i \varphi(x_i) h_i^T
\end{cases}
$$

# From KPCA to RKM representation (3)

- Elimination of $W$ gives the eigenvalue decomposition:

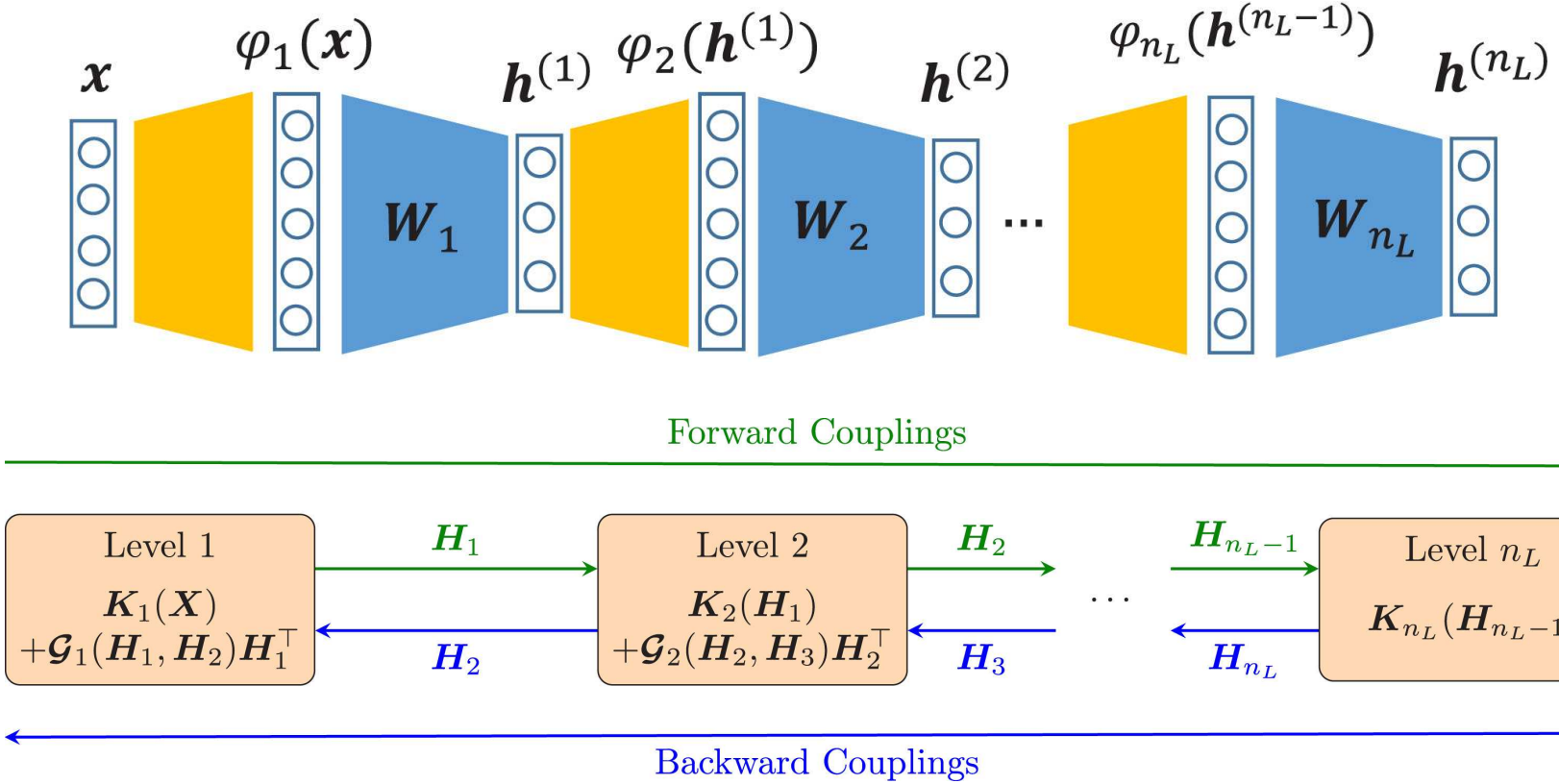
$$\frac{1}{\eta} K H^T = H^T \Lambda$$

where $H = [h_1 ... h_N] \in \mathbb{R}^{s \times N}$ and $\Lambda = \mathrm{diag}\{\lambda_1, ..., \lambda_s\}$ with $s \leq N$

- Primal and dual model representations

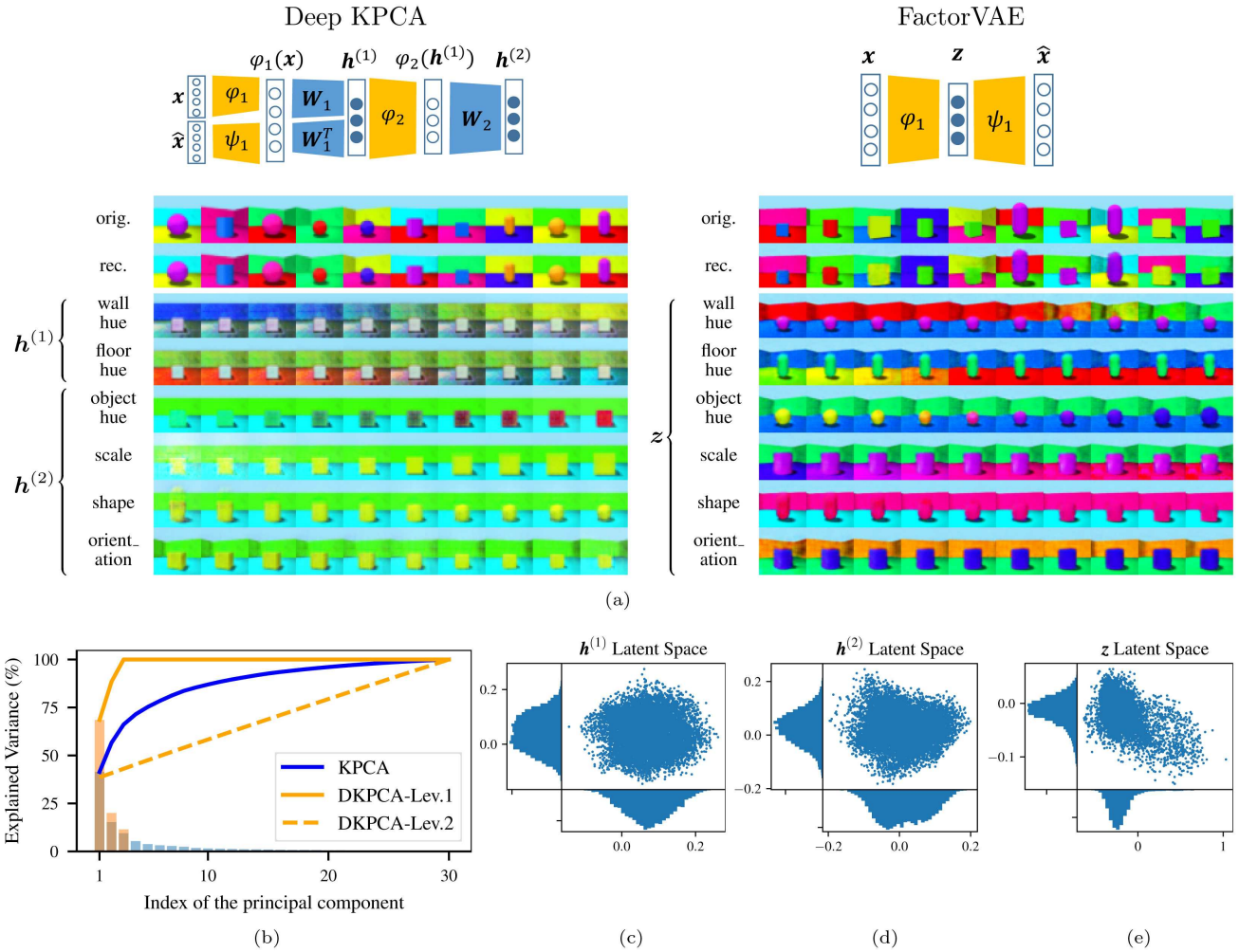
$$\mathcal{M} \begin{cases} \nearrow & (P)_{\mathrm{RKM}}: \quad \hat{e} = W^T \varphi(x) \\ \searrow & (D)_{\mathrm{RKM}}: \quad \hat{e} = \frac{1}{\eta} \sum_j h_j K(x_j, x). \end{cases}$$

# Deep KPCA based on RKM representations



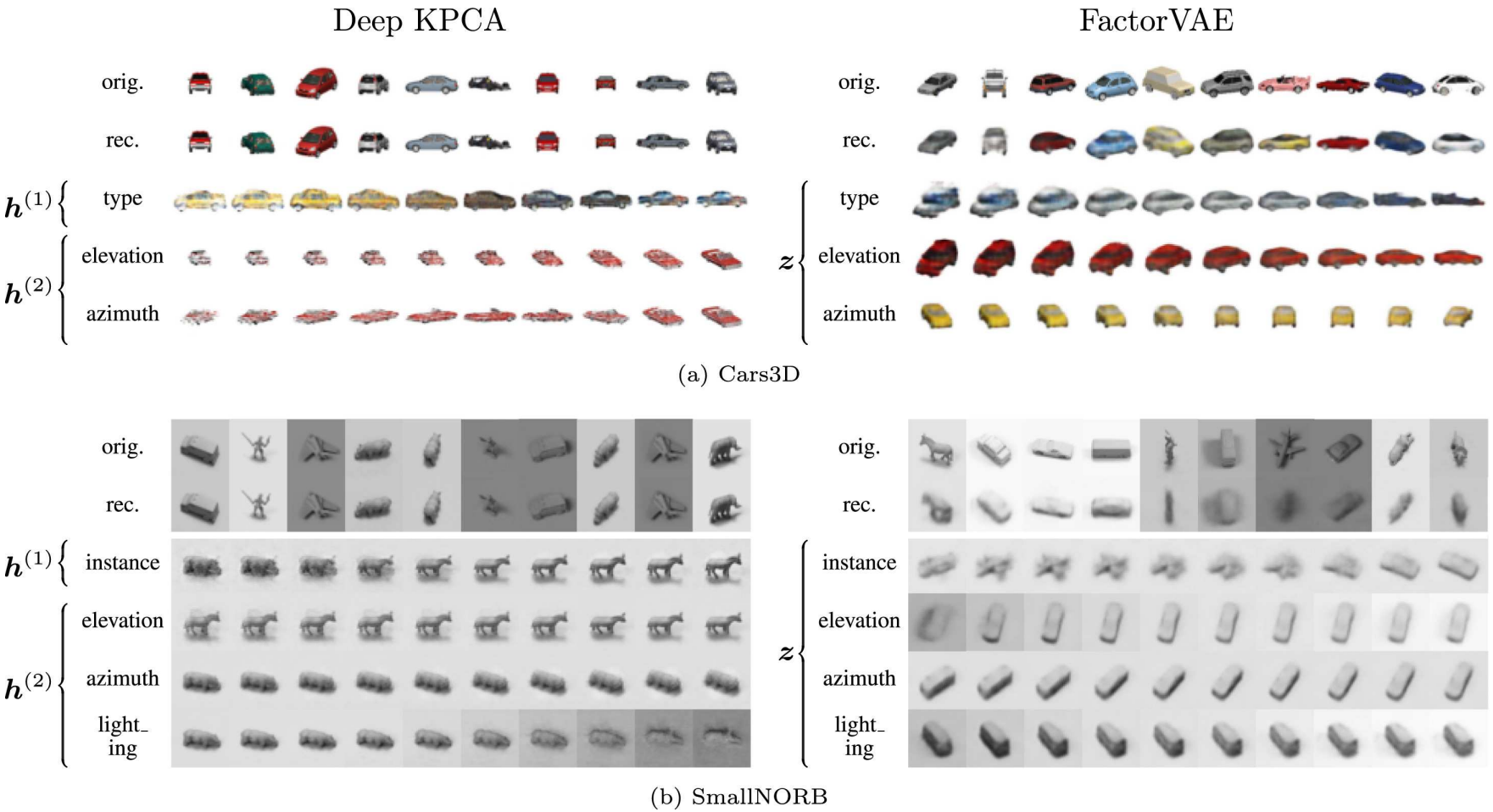

[Tonin, Tao, Patrinos & Suykens, Neural Networks 2024]

# Towards "white box": deep eigenvalues/eigenvectors (1)




[Tonin, Tao, Patrinos & Suykens, Neural Networks 2024]

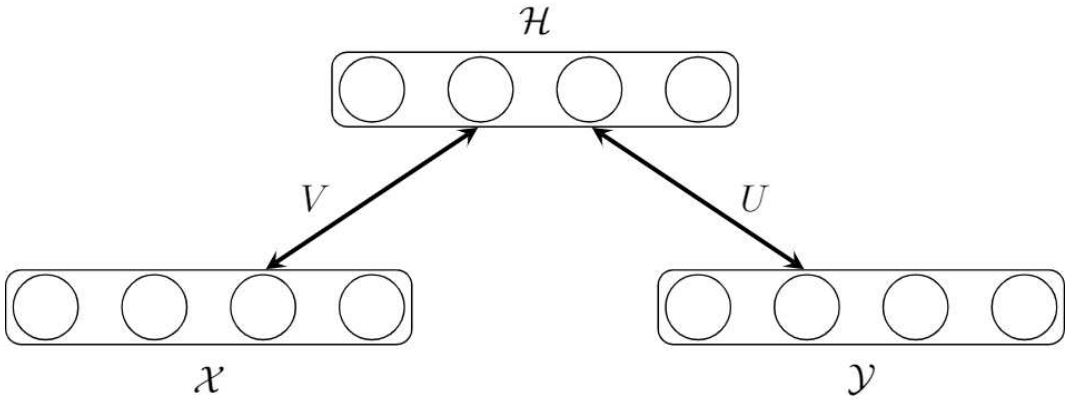
# Towards "white box": deep eigenvalues/eigenvectors (2)



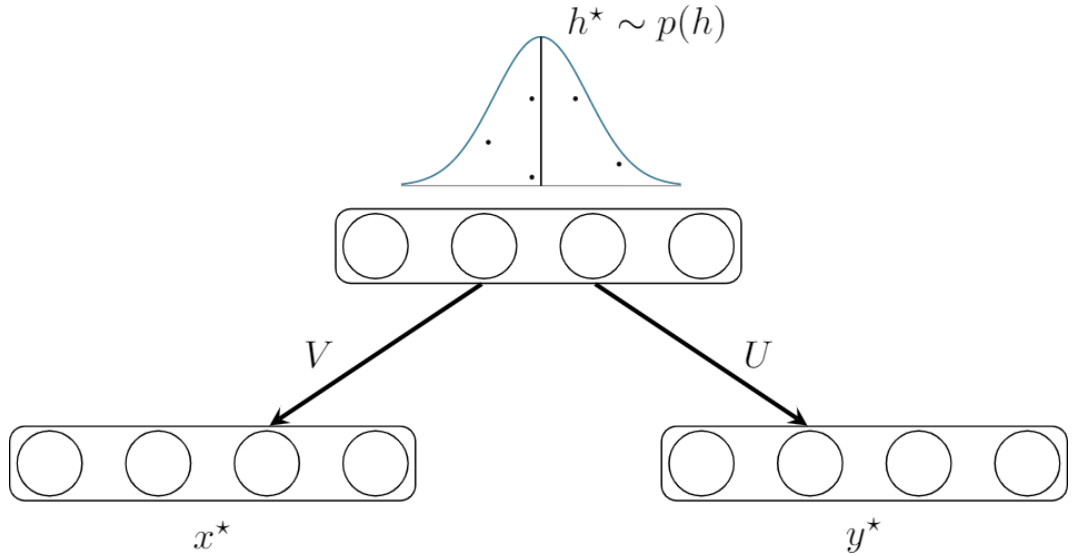

[Tonin, Tao, Patrinos & Suykens, Neural Networks 2024]

# Multi-view Generative RKM (Gen-RKM)

**Train:**

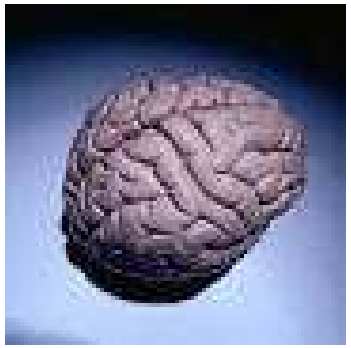



**Generate:**

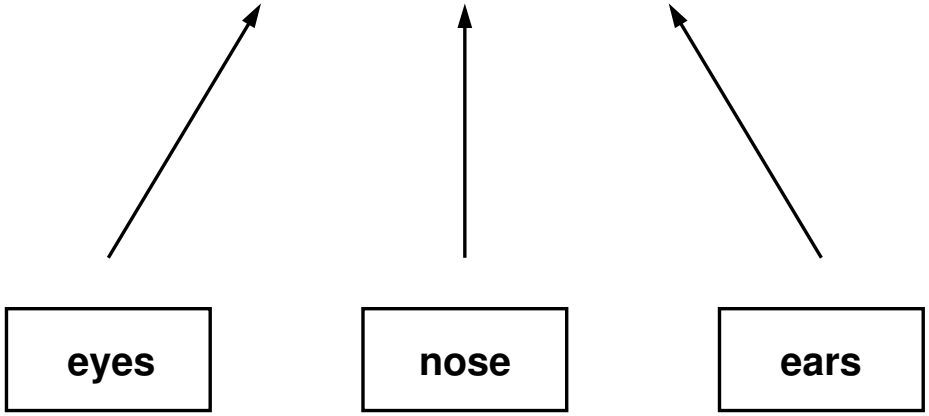



[Pandey, Schreurs & Suykens, Neural Networks, 2021]

# Analogy with the human brain



*shared feature representation*




*multimodal data*

# Multi-view Gen-RKM

The objective

$$J_{\text{train}}(\boldsymbol{h}_i, \boldsymbol{U}, \boldsymbol{V}) = -\sum_{i=1}^{N} \phi_1(\boldsymbol{x}_i)^T \boldsymbol{U} \boldsymbol{h}_i - \sum_{i=1}^{N} \phi_2(\boldsymbol{y}_i)^T \boldsymbol{V} \boldsymbol{h}_i + \frac{\lambda}{2} \sum_{i=1}^{N} \boldsymbol{h}_i^T \boldsymbol{h}_i + \frac{\eta_1}{2} \text{Tr}(\boldsymbol{U}^T \boldsymbol{U}) + \frac{\eta_2}{2} \text{Tr}(\boldsymbol{V}^T \boldsymbol{V})$$

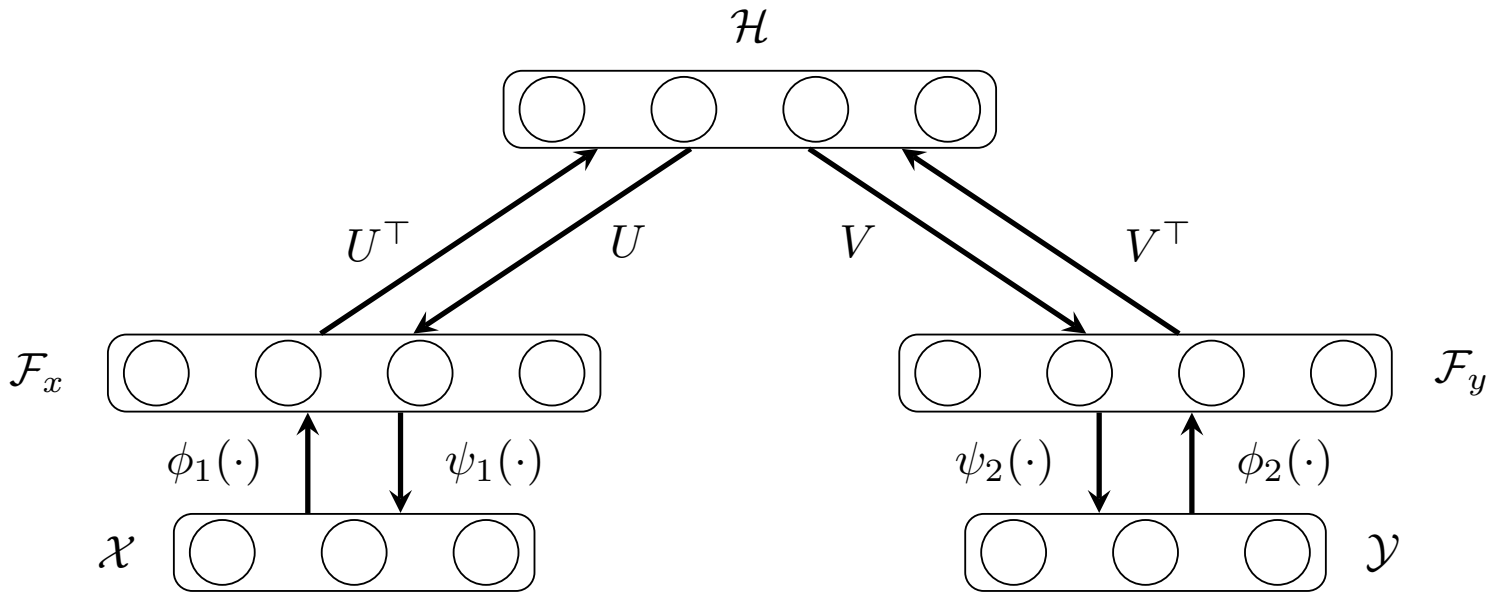
results for **training** into the eigenvalue problem

$$(\frac{1}{\eta_1} \boldsymbol{K}_1 + \frac{1}{\eta_2} \boldsymbol{K}_2) \boldsymbol{H}^T = \boldsymbol{H}^T \boldsymbol{\Lambda}$$

with $\boldsymbol{H} = [\boldsymbol{h}_1 ... \boldsymbol{h}_N]$ and kernel matrices $\boldsymbol{K}_1, \boldsymbol{K}_2$ related to $\phi_1, \phi_2$.

[Pandey, Schreurs & Suykens, Neural Networks, 2021

# Multi-view Gen-RKM




Gen-RKM schematic representation modeling a common subspace $\mathcal{H}$ between two data sources $\mathcal{X}$ and $\mathcal{Y}$. The $\phi_1$, $\phi_2$ are the feature maps ($\mathcal{F}_x$ and $\mathcal{F}_y$ represent the feature-spaces) corresponding to the two data sources. While $\psi_1$, $\psi_2$ represent the pre-image maps. The interconnection matrices $U, V$ model dependencies between latent variables and the mapped data sources.

[Pandey, Schreurs & Suykens, Neural Networks, 2021

# Gen-RKM: using an explicit (CNN) feature map

Parametrized feature maps: $\phi_{\boldsymbol{\theta}}(\cdot)$, $\psi_{\boldsymbol{\zeta}}(\cdot)$ (e.g. CNN and transposed CNN).

Overall objective function, using a stabilization mechanism [Suykens, 2017]:

$$\begin{aligned}\min_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\zeta}_1,\boldsymbol{\zeta}_2} \mathcal{J}_c \;=\; & \mathcal{J}_{\text{train}} + \tfrac{c_{\text{stab}}}{2}\mathcal{J}_{\text{train}}^2 \\ & + \tfrac{c_{\text{acc}}}{2N}\left(\sum_{i=1}^{N}\left[\mathcal{L}_1(\boldsymbol{x}_i^\star, \psi_{1_{\boldsymbol{\zeta}_1}}(\phi_{1_{\boldsymbol{\theta}_1}}(\boldsymbol{x}_i^\star))) + \mathcal{L}_2(\boldsymbol{y}_i^\star, \psi_{2_{\boldsymbol{\zeta}_2}}(\phi_{2_{\boldsymbol{\theta}_2}}(\boldsymbol{y}_i^\star)))\right]\right)\end{aligned}$$

with reconstruction errors

$$\begin{aligned}\mathcal{L}_1(\boldsymbol{x}_i^\star, \psi_{1_{\boldsymbol{\zeta}_1}}(\phi_{1_{\boldsymbol{\theta}_1}}(\boldsymbol{x}_i^\star))) &= \tfrac{1}{N}\|\boldsymbol{x}_i^\star - \psi_{1_{\boldsymbol{\zeta}_1}}(\phi_{1_{\boldsymbol{\theta}_1}}(\boldsymbol{x}_i^\star))\|_2^2, \\ \mathcal{L}_2(\boldsymbol{y}_i^\star, \psi_{2_{\boldsymbol{\zeta}_2}}(\phi_{2_{\boldsymbol{\theta}_2}}(\boldsymbol{y}_i^\star))) &= \tfrac{1}{N}\|\boldsymbol{y}_i^\star - \psi_{2_{\boldsymbol{\zeta}_2}}(\phi_{2_{\boldsymbol{\theta}_2}}(\boldsymbol{y}_i^\star))\|_2^2\end{aligned}$$

and with $\Phi_{\boldsymbol{x}} = [\phi_1(\boldsymbol{x}_1), \ldots, \phi_1(\boldsymbol{x}_N)], \Phi_{\boldsymbol{y}} = [\phi_2(\boldsymbol{y}_1), \ldots, \phi_2(\boldsymbol{y}_N)]$, $U, V$ from

$$\begin{bmatrix} \frac{1}{\eta_1}\Phi_{\boldsymbol{x}}\Phi_{\boldsymbol{x}}^\top & \frac{1}{\eta_1}\Phi_{\boldsymbol{x}}\Phi_{\boldsymbol{y}}^\top \\ \frac{1}{\eta_2}\Phi_{\boldsymbol{y}}\Phi_{\boldsymbol{x}}^\top & \frac{1}{\eta_2}\Phi_{\boldsymbol{y}}\Phi_{\boldsymbol{y}}^\top \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} U \\ V \end{bmatrix} \Lambda.$$

# Gen-RKM: latent space exploration



CelebA reconstructed images by bilinear-interpolation in latent space
[Pandey, Schreurs & Suykens, Neural Networks 2021]

# Gen-RKM - traversals along principal components

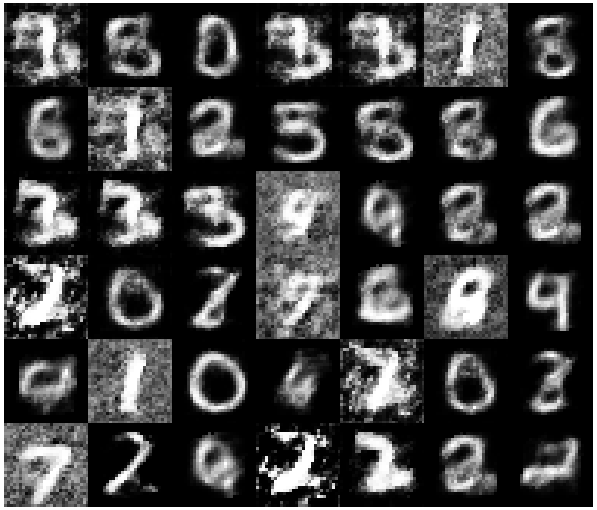



Disentangled Representation Learning and Generation with Manifold Optimization

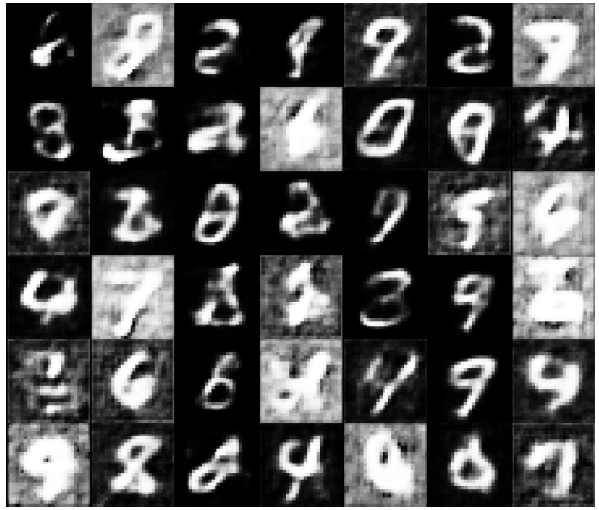
[Pandey, Fanuel, Schreurs & Suykens, Neural Computation, 2022]

**Software:** see https://www.esat.kuleuven.be/stadius/E/software.php

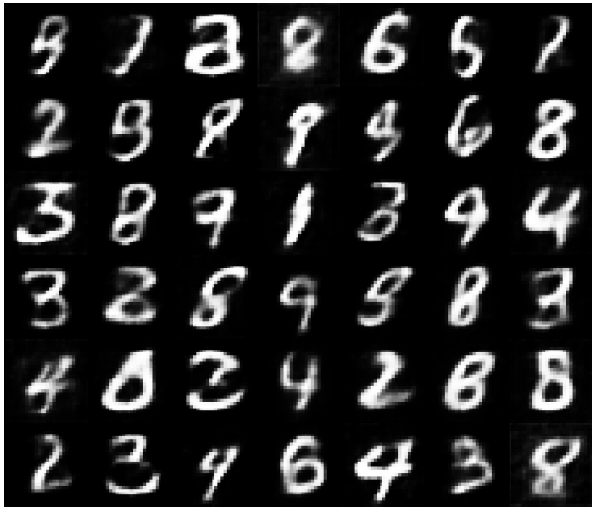
# Robust Gen-RKM - Robust generation



VAE



Gen-RKM



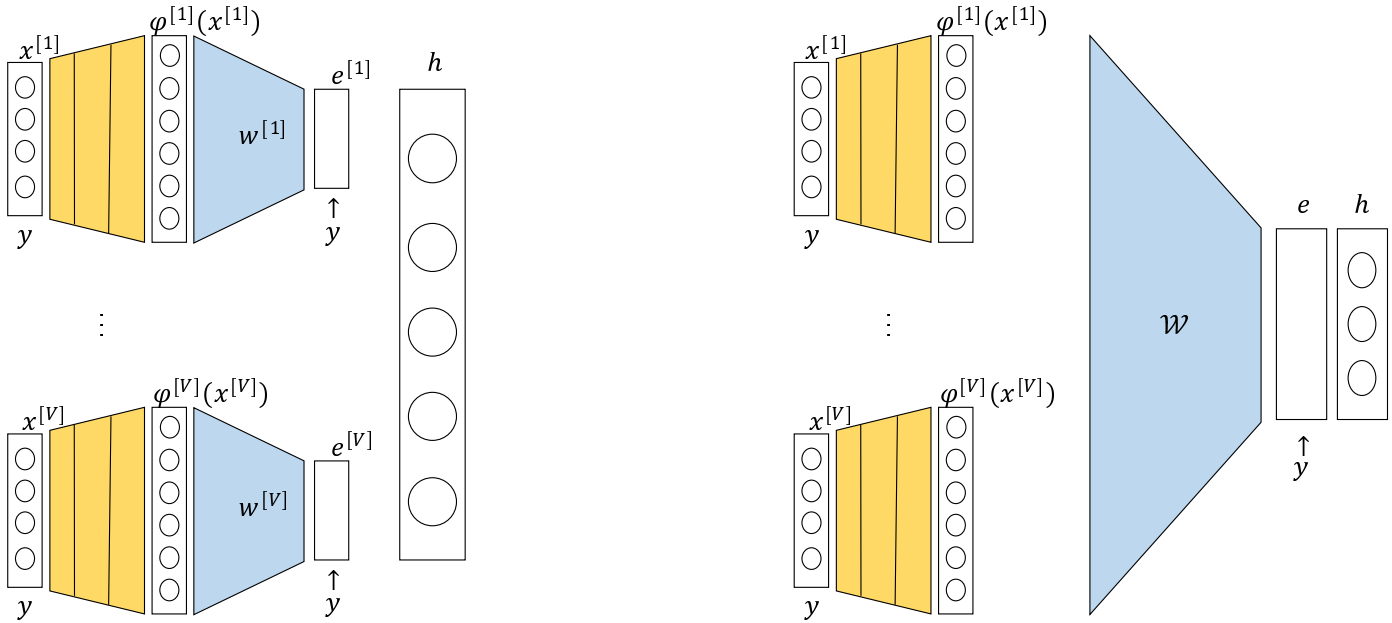
Robust Gen-RKM

Weighted conjugate feature duality: $\frac{1}{2\lambda}e^T De + \frac{\lambda}{2}h^T D^{-1}h \geq e^T h$

[Pandey, Schreurs & Suykens, LOD 2020]

# Tensor-based RKM for Multi-view KPCA

$$\min \langle \mathcal{W}, \mathcal{W} \rangle - \sum_{i=1}^{N} \left\langle \Phi_{(i)}, \mathcal{W} \right\rangle h_i + \lambda \sum_{i=1}^{N} h_i^2 \quad \text{with} \quad \Phi_{(i)} = \varphi^{[1]}(x_i^{[1]}) \otimes \ldots \otimes \varphi^{[V]}(x_i^{[V]})$$




[Houthuys & Suykens, ICANN 2018]

# Recurrent RKM

- Objective for **Recurrent RKM** [Pandey et al., 2022]:

$$J = -\sum_t \varphi(x_t)^T W h_t - \sum_t \sum_l h_{t-l}^T h_t + \frac{1}{2} h_t^T \Lambda h_t + \frac{1}{2} \mathrm{Tr} W^T W$$

- Results into a sequence $\{h_t\}$ in the latent space from which the future of the time-series is predicted.

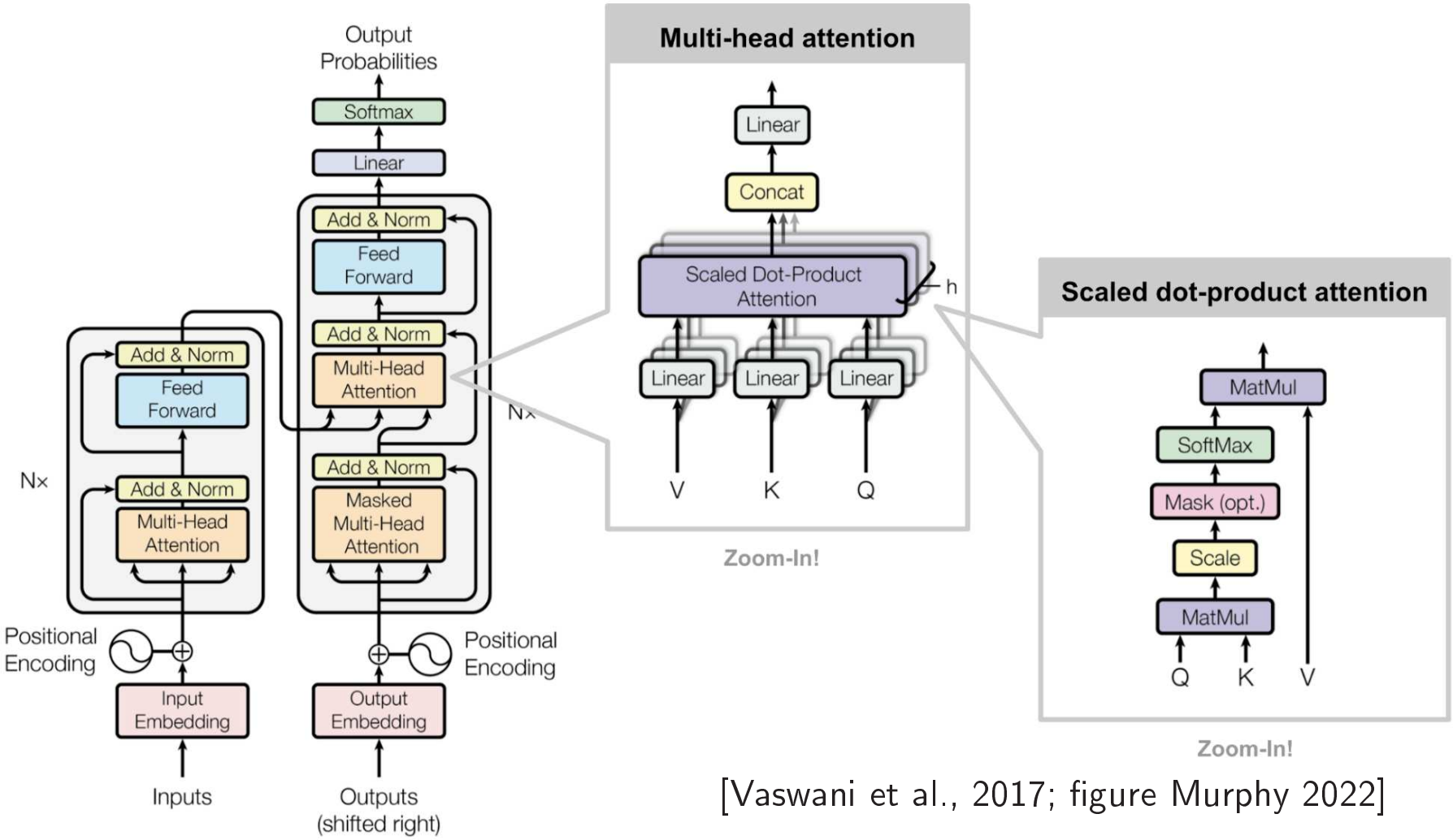
- $\{h_t\}$ follows from the solution to an eigenvalue problem.

- Related work on recurrent temporal RBM [Sutskever et al., 2008; Osogami, 2019]

[Pandey, De Meulemeester, De Plaen, De Moor, Suykens, ESANN 2022]

# *Self-attention and transformers within the LS-SVM framework*

# Transformers




[Vaswani et al., 2017; figure Murphy 2022]

# SVD within the LS-SVM setting

- **Primal problem** (new variational principle):

$$\min_{w,v,e,r} \; -w^T v + \tfrac{1}{2}\gamma \sum_{i=1}^N e_i^2 + \tfrac{1}{2}\gamma \sum_{j=1}^M r_j^2 \text{ subject to } \quad \begin{array}{l} e_i = w^T\varphi(x_i),\ i = 1,...,N \\ r_j = v^T\psi(z_j),\ j = 1,...,M \end{array}$$

  with $e_i, r_j \in \mathbb{R}$ and $w, v \in \mathbb{R}^N$, and $\varphi, \psi$ compatible feature maps

- Lagrangian $\mathcal{L} = J - \sum_i \alpha_i \left(e_i - w^T\varphi(x_i)\right) - \sum_j \beta_j \left(r_j - v^T\psi(z_j)\right)$

- Optimality conditions $\frac{\partial \mathcal{L}}{\partial w} = 0, \frac{\partial \mathcal{L}}{\partial v} = 0, \frac{\partial \mathcal{L}}{\partial e_i} = 0, \frac{\partial \mathcal{L}}{\partial r_j} = 0, \frac{\partial \mathcal{L}}{\partial \alpha_i} = 0, \frac{\partial \mathcal{L}}{\partial \beta_j} = 0$

- Eliminate $w, v, e, r$; collect solutions corresponding to non-zero $\lambda = \frac{1}{\gamma}$:

$$\begin{bmatrix} 0 & K \\ K^T & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \lambda \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \text{ with kernel matrix } K_{ij} = \varphi(x_i)^T\psi(z_j).$$

[Suykens, ACHA 2016]

# Primal and dual model representations

Primal and dual model representations

$$\mathcal{M} \begin{cases} \nearrow & (P): \quad e_i = w^T \varphi(x_i) \\ & \qquad\quad r_j = v^T \psi(z_j) \\ \searrow & (D): \quad e_i = \sum_j \beta_j \psi(z_j)^T \varphi(x_i) = \sum_j \beta_j K(z_j, x_i) \\ & \qquad\quad r_j = \sum_i \alpha_i \varphi(x_i)^T \psi(z_j) = \sum_i \alpha_i K(x_i, z_j) \end{cases}$$

[Suykens, ACHA 2016]

# Kernel PCA versus Kernel SVD

- **Kernel PCA:** one data source $\{x_i\}, i = 1, ..., N$:

$$K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$$

with kernel $K$ symmetric and positive definite, related to feature map $\phi$.

- **Kernel SVD:** two data sources $\{x_i\}, i = 1, ..., N$ and $\{z_j\}, j = 1, ..., M$:

$$K(x_i, z_j) = \varphi(x_i)^T \psi(z_j)$$

with kernel $K$ asymmetric, related to compatible feature maps $\phi, \psi$.

# Self-attention with Asymmetric Kernel

Let $\{x_i \in \mathbb{R}^d\}_{i=1}^N$ be the input data sequence.
In **self-attention** [Vaswani et al., 2017] **queries**, **keys** and **values** are

$$q(x_i) = W_q x_i, \;\; k(x_i) = W_k x_i, \;\; v(x_i) = W_v x_i,$$

where $W_q \in \mathbb{R}^{d_q \times d}, W_k \in \mathbb{R}^{d_k \times d}, W_v \in \mathbb{R}^{d_v \times d}$ (commonly $d_q = d_k$).
Attention scores: $a(x_i, x_j) = \langle q(x_i), k(x_j) \rangle / \sqrt{d_k} = \langle W_q x_i, W_k x_j \rangle / \sqrt{d_k}$.
Canonical self-attention:

$$\kappa(x_i, x_j) = \mathrm{softmax}(\langle q(x_i), k(x_j) \rangle) / \sqrt{d_k}, \;\; i, j = 1, ..., N$$

which is an **asymmetric kernel function**, i.e. $\kappa(x_i, x_j) \neq \kappa(x_j, x_i)$.
The attention output $o_i \in \mathbb{R}^{d_v}$ in each head is

$$o_i = \sum_{j=1}^{N} v(x_j) \kappa(x_i, x_j) = \sum_{j=1}^{N} v(x_j) K_{ij}, \quad i = 1, ..., N.$$

# Primal-dual representations for self-attention

- **Self-attention** primal problem [Chen, Tao, Tonin, Suykens, NeurIPS 2023]

$$\begin{aligned} \max_{W_e, W_r, e_i, r_j} \quad & J = \frac{1}{2}\sum_{i=1}^{N} e_i^T \Lambda e_i + \frac{1}{2}\sum_{j=1}^{N} r_j^T \Lambda r_j - \mathrm{Tr}(W_e^T W_r) \\ s.t. \quad & e_i = (f(X)^T W_e)^T \phi_q(x_i), \ i = 1, ..., N, \\ & r_j = (f(X)^T W_r)^T \phi_k(x_j), \ j = 1, ..., N \end{aligned}$$

where $\phi_q, \phi_k$ are **feature maps** related to queries and keys, respectively, $\Lambda$ diagonal matrix with positive diagonal elements. The dual problem gives a shifted eigenvalue problem, related to **Kernel SVD**.

- In **primal-attention**, the attention outputs are

$$o_i = [e_i; r_i] = [W_e^T f(X) g_q(q(x_i)); W_r^T f(X) g_k(k(x_i))]$$

where $K_{ij} = \langle g_q(q(x_i)), g_k(k(x_j)\rangle = \langle \phi_q(x_i), \phi_k(x_j)\rangle$.

# Training of Transformers

Training of **Transformers** based on **primal-attention** [Chen et al., 2023]:

$$\min L + \eta \sum_l J_l^2$$

with $\eta > 0$, $L$ a task-oriented loss (e.g. cross-entropy for classification tasks) and $J_l$ are the primal objectives of the several attention blocks with the primal-attention scheme. Here each $J_l$ is of the form

$$J = \frac{1}{2}\sum_{i=1}^{N} e_i^T \Lambda e_i + \frac{1}{2}\sum_{j=1}^{N} r_j^T \Lambda r_j - \mathrm{Tr}(W_e^T W_r).$$

It exploits the property that for all components evaluated in the primal objective $J$ one has $J = 0$ (hence the term $J_l^2$ in the objective for training the Transformer). In this way **low-rank representations** can be obtained, together with **efficient training in the primal**.

# Kernel SVD extension to multiple data sources

- Primal problem:

$$\begin{aligned} \max_{w_q, e_{i_q}^{(q)}} \quad & J = (\frac{Q}{2} - 1) \sum_{q=1}^{Q} w_q^T w_q - \sum_{1 \leq p < q \leq Q} w_q^T w_p + \frac{Q-1}{2} \gamma \sum_{q=1}^{Q} \sum_{i_q=1}^{N_q} (e_{i_q}^{(q)})^2 \\ \text{subject to} \quad & e_{i_q}^{(q)} = w_q^T \phi_q(x_{i_q}^{(q)}), \ \ \forall i_q, \forall q \end{aligned}$$

- Dual:

$$\begin{bmatrix} 0 & K_{12} & \cdots & K_{1Q} \\ K_{21} & 0 & & \vdots \\ \vdots & & \ddots & K_{(Q-1)Q} \\ K_{Q1} & \cdots & K_{Q(Q-1)} & 0 \end{bmatrix} \begin{bmatrix} \alpha^{(1)} \\ \vdots \\ \alpha^{(Q)} \end{bmatrix} = \frac{1}{\gamma} \begin{bmatrix} \alpha^{(1)} \\ \vdots \\ \alpha^{(Q)} \end{bmatrix}$$

[Zeng, Tao, Suykens, arXiv: 2610.03216, 2026]

# Future Challenges and Conclusions

- LS-SVM as **core models** in supervised and unsupervised learning, and beyond
- **primal and dual** model representations, **parametrized feature maps** or **kernel-based**
- combining powerful deep learning models with kernel-based setting
- LS-SVM framework: **new connections with Transformers, RBMs, Deep BM, VAE**
- new schemes for **generative models** and **deep kernel machines**
- **multi-view** and **tensor-based** models
- robustness, disentanglement, latent space exploration

# Acknowledgements (1)


- Current and former co-workers at ESAT-STADIUS:

  S. Achten, C. Alzate, M. Campling, Y. Chen, J. De Brabanter, K. De Brabanter, B. De Cooman, L. De Lathauwer, H. De Meulemeester, B. De Moor, H. De Plaen, Ph. Dreesen, M. Espinoza, T. Falck, M. Fanuel, Y. Feng, B. Gauthier, B. Hamers, F. He, L. Hoegaerts, X. Huang, L. Houthuys, V. Jumutc, Z. Karevan, A. Kumari, A. Lambert, R. Langone, F. Liu, L. Lukas, R. Mall, S. Mehrkanoon, M. Orchel, A. Pandey, P. Patrinos, K. Pelckmans, S. RoyChowdhury, S. Salzo, P. Sanchez Garcia, J. Schreurs, M. Signoretto, Q. Tao, F. Tonin, J. Vandewalle, T. Van Gestel, S. Van Huffel, C. Varon, L. Viano, D. Winant, Y. Yang, X. Zeng and others

- Many other people for joint work, discussions, invitations, organizations

- Support from ERC AdG E-DUALITY, ERC AdG A-DATADRIVE-B, KU Leuven, OPTEC, DYSCO, FWO, IWT, Flanders AI, Leuven.AI

# Acknowledgements (2)









# Acknowledgements (3)




ERC Advanced Grant E-DUALITY

Exploring duality for future data-driven modelling

# Thank you